John von Neumann Institute for Computing

Efficient Methods for Long Range Interactions in Periodic Geometries Plus One Application

Christian Holm

published in

Computational Soft Matter: From Synthetic Polymers to Proteins, Lecture Notes,
Norbert Attig, Kurt Binder, Helmut Grubmüller, Kurt Kremer (Eds.),
John von Neumann Institute for Computing, Jülich,
NIC Series, Vol. **23***, ISBN 3-00-012641-4, pp. 195-236, 2004.*

© 2004 by John von Neumann Institute for Computing
Permission to make digital or hard copies of portions of this work for personal or classroom use is granted provided that the copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise requires prior specific permission by the publisher mentioned above.

http://www.fz-juelich.de/nic-series/volume23

Efficient Methods for Long Range Interactions in Periodic Geometries Plus One Application

Christian Holm

Max-Planck-Institut für Polymerforschung,
55021 Mainz, Germany
E-mail: holm@mpip-mainz.mpg.de
http://www.espresso.mpg.de/

We give an extensive introduction into the topic of how to compute efficiently long range interactions. We start out by reviewing the traditional Ewald sum for 3D Coulomb systems, discuss then in some detail the P^3M method of Hockney and Eastwood, and briefly mention alternative ways of dealing with the Coulomb sum. We then discuss the best strategies to perform the sum under partially periodic boundary conditions. We also review the dipolar Ewald sum. For all our methods we provide error formulas which enable us to tune the algorithm at predefined accuracy. We then present the structure and design of our newly developed program package ESPResSo which contains already many of the presented methods. At the end we provide results of an extensive Molecular Dynamics simulations where we study the behavior of polyelectrolytes in poor solvents, taking explicitely care of the counterions. The conformations show pearl-necklace structures that are subject to strong conformational fluctuations, leading only to small signatures in the form factor. In addition we study how the necklace collapses as a function of Bjerrum length. At last we demonstrate that the position of the first peak in the inter-chain structure factor varies with the monomer density close to $\rho_m^{1/3}$ for all densities which is in strong contrast to polyelectrolyte solutions in good solvent.

1 Introduction

Soft condensed matter (or soft matter, as it is often called) is a term for materials in states of matter that are neither simple liquids nor hard solids of the type studied, for example, in solid state physics. Many such materials are familiar from everyday life - glues, paints, soaps, baby diapers - while others are important in industrial processes, such as polymer melts that are molded and extruded to form plastics[1]. Biological materials are mainly made out of soft matter as well - membranes, actin filaments, DNA, RNA, and proteins belong to this class. Furthermore, most of the food we digest is soft matter. All these materials share the importance of length scales intermediate between atomic and macroscopic scales: The relevant range for soft matter lies between nanometers and micrometers. Examples are polymers, colloids, liquid crystals, glasses, and dipolar fluids. Typical energies between different structures are similar to thermal energies. Hence, Brownian motion or thermal fluctuations play a prominent role. Another key feature of soft matter systems is their propensity to self-assemble. Again the energy differences during this process are small such that many neighboring states are normally accessible through fluctuations. This often results in complex phase behaviors yielding a rich variety of accessible structures. Order does not necessarily arise on the single molecule level, but quite commonly exhibits a multitude of hierarchically ordered structures of sometimes tremendous intricacy and complexity. Most of the biological systems are usually not even in equilibrium but evolve among switchable steady states.

Given this wide field, research on soft material substances often acquires knowledge from different areas of research, such as physics, chemistry, and biology, such that a high level of interdisciplinary may be required for certain scientific questions.

In the past, our research has mainly focused on the study of charged polymers (polyelectrolytes) and charged colloids which serve as important substances for many technical applications. Charged systems also occur in biological environments (since most biological matter is charged)[2], and modeling explicit water molecules requires partial charges as well.

Computational simulations provide some unique ways to elucidate the properties of charged systems. However, they are time consuming, since the interaction between charged species is long ranged. Moreover, to compute the interactions of N particles fast and accurately, one needs smart algorithms to beat the unfavorable $\mathcal{O}(N^2)$ complexity which one obtains from simply counting all particle interactions.

In this contribution we give a short overview over recently developed methods to treat Coulomb interactions in fully or partially periodic geometries, and also show how to treat dipolar interactions. Next we present some features of our newly designed program package ESPResSo which implements most of these algorithms. We conclude with a short overview of an application to polyelectrolytes in poor solvent.

2 Methods for the Coulomb Sum in 3D

One of the biggest problems for the simulations of charged systems is the long range nature of the Coulomb interaction. In principle, each charge interacts with all others, leading to a computational effort of $\mathcal{O}(N^2)$ already within the central simulation box. For many physical investigations one wants to simulate bulk properties and therefore introduces periodic boundary conditions in all spatial directions to avoid boundary effects. The famous Ewald sum[3–6] does a remarkable job in splitting the very slowly converging sum over the Coulomb potential into two exponentially converging sums. Still, this method suffers from two deficiencies. First, it is computationally demanding. This is due to the fact that one part of the problem is solved in reciprocal space implying the need for several Fourier transformations. Second, the algorithm scales like N^2 with N being the number of charged particles in the primary box, or at best like $N^{3/2}$, if one uses cutoffs which are optimized with respect to the splitting parameter[7]. The situation becomes worse if only partially periodic boundary conditions are applied. In this section we give an introduction to the Ewald summation, collecting the important equations for energy and forces. We discuss briefly some alternative methods. Some recent results of our research on how to deal with partially periodic boundary conditions will be deferred to the next two sections. The material has been mainly collected from the sources[8–13]. As an additional good textbook we can recommend the second edition of Frenkel and Smit[14].

2.1 The Standard 3D Ewald Method

There are many examples of long-range interactions which can be treated by Ewald techniques[15]. Here, however, only the case of Coulomb point charges will be discussed, i.e., an interaction potential $1/r$. Consider therefore a system of N particles with charges q_i at positions $\mathbf{r}_i$ in an overall neutral and, for simplicity, cubic simulation box of length L. If

periodic boundary conditions are applied, the total electrostatic energy of the box is given by

$$E = \frac{1}{2} \sum_{\mathbf{n}\in\mathbb{Z}^3}{}' \sum_{i,j=1}^{N} \frac{q_i q_j}{|\mathbf{r}_{ij} + \mathbf{n}L|}, \tag{1}$$

where $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$. Since this sum is only conditionally convergent, its value is not well-defined unless one specifies the precise way in which the cluster of simulation boxes should fill the $\mathbb{R}^3$, i.e., its *shape*, e.g. approximately spherical. A more detailed discussion of these points can be found in Refs.[5,6,16,17].

It is often stated that the long-range nature of the Coulomb potential complicates the treatment of electrostatic interactions. This is, however, only one part of the problem. In fact, the Coulomb potential bears *two* intrinsic difficulties. It is slowly decaying at large distances, and strongly varying at small distances. It is the combination of these two properties which leads to severe problems. If only one of them was present, everything would be comparatively easy, since a short-range potential could be treated by a simple cutoff, as it is done, e.g., for interactions of the Lennard-Jones type, and a long-range potential, which is periodic and slowly varying *everywhere*, can accurately be represented by the first few terms of its Fourier series.

Obviously, each of the two complications forbids the simple solution of the other, and the slowly decaying long-range part of the Coulomb potential renders a straightforward summation of Eqn. (1) impracticable. The trick is thus to split the problem into two parts by the trivial identity

$$\frac{1}{r} = \frac{f(r)}{r} + \frac{1 - f(r)}{r}. \tag{2}$$

The underlying idea is to distribute the two complications between the two terms in Eqn. (2) by a suitable choice of the splitting function f. In particular:

- The first part $\frac{f(r)}{r}$ should be negligible, or even zero, beyond some cutoff $r_{\max}$, so that the summation up to the cutoff is a good approximation to (or the exact result of) this contribution to the total electrostatic potential.

- The second part $\frac{1-f(r)}{r}$ should be a slowly varying function for *all* r, so that its Fourier transform can be represented by only a few $\mathbf{k}$-vectors with $|\mathbf{k}| \leq k_{\max}$. This permits an efficient calculation of this contribution to the total electrostatic potential in reciprocal space.

Since the field equations are linear, the sum of these two contributions gives the solution for the potential of the original problem.

The two requirements on the splitting function f mentioned above leave a large freedom of choice[4,18,19]. The traditional selection is the complementary error function $\mathrm{erfc}(r) := \frac{2}{\sqrt{\pi}} \int_r^\infty \mathrm{d}t \ \mathrm{e}^{-t^2}$. This results in the well known Ewald formula for the electrostatic energy of the primary box:

$$E = E^{(r)} + E^{(k)} + E^{(s)} + E^{(d)}, \tag{3}$$

where $E^{(r)}$ is the contribution from real space, $E^{(k)}$ the contribution from reciprocal space, $E^{(s)}$ the self energy and $E^{(d)}$ the dipole correction. They can be written as[14,15]

$$E^{(r)} = \frac{1}{2} \sum_{\mathbf{m}\in\mathbb{Z}^3}{}' \sum_{i,j} q_i q_j \frac{\operatorname{erfc}(\alpha|\mathbf{r}_{ij} + \mathbf{m}L|)}{|\mathbf{r}_{ij} + \mathbf{m}L|} \tag{4}$$

$$E^{(k)} = \frac{1}{2}\frac{1}{V} \sum_{\mathbf{k}\neq\mathbf{0}} \frac{4\pi}{k^2} e^{-k^2/4\alpha^2} |\tilde{\rho}(\mathbf{k})|^2 \tag{5}$$

$$E^{(s)} = -\frac{\alpha}{\sqrt{\pi}} \sum_i q_i^2 \tag{6}$$

$$E^{(d)} = \frac{2\pi}{(1+2\epsilon')V} \left(\sum_i q_i \mathbf{r}_i \right)^2, \tag{7}$$

where the Fourier transformed charge density $\tilde{\rho}(\mathbf{k})$ is defined as

$$\tilde{\rho}(\mathbf{k}) = \int_V \mathrm{d}^3 r \; \rho(\mathbf{r})\, e^{-\mathrm{i}\,\mathbf{k}\cdot\mathbf{r}} = \sum_{j=1}^{N} q_j \; e^{-\mathrm{i}\,\mathbf{k}\cdot\mathbf{r}_j} \qquad \text{where} \qquad \mathbf{k} \in \frac{2\pi}{L}\mathbb{Z}^3. \tag{8}$$

The advantage of rewriting Eqn. (1) this way is that the exponentially converging sums over $\mathbf{m}$ and $\mathbf{k}$ in (4,5) allow the introduction of comparatively small cutoffs without much loss in accuracy. Typically one chooses α large enough as to employ the minimum image convention in Eqn. (4). The inverse length α, which is often referred to as the Ewald (or splitting) parameter, tunes the relative weights of the real space and the reciprocal space contributions. However, the final result of the exact equation (3), not terminating the sums at some finite cutoff value, is independent of α. The form (7) given for the dipole correction assumes that the set of periodic replications of the simulation box tends spherically towards an infinite cluster and that the medium outside this sphere is a homogeneous dielectric with dielectric constant ϵ'[5], whereas inside we set for simplicity $\epsilon = 1$. The derivation of this term is not straightforward and requires an accurate mathematical treatment of the conditional convergence of Eqn. (1), and of the image charges generated in the corresponding dielectric medium. Note that the case of a surrounding vacuum corresponds to $\epsilon' = 1$ and that the dipole correction vanishes for metallic (or "tinfoil") boundary conditions, since then $\epsilon' = \infty$. A detailed discussion of this term can be found in Ref.[5,6,17]. The dipole correction in Eqn. (7) is independent of the Ewald parameter α. This again shows that the correction is not specific to the Ewald sum but more generally reflects the problems inherent to the conditional convergence of the $\mathbf{n}$-sum in Eqn. (1). For the computation of $E^{(d)}$ the particle coordinates must *not* be folded back into the primary unit cell, for otherwise each boundary crossing produces an unphysical jump in the electrostatic energy, see Ref.[16]. The cutoffs $r_{\max}$ and $k_{\max}$ can be optimized with respect to α such that the required computer time scales like $N^{3/2}$, see Perram et al.[7]. This, however, may require that $r_{\max} > L/2$ prohibiting the simple minimum image convention in real space and rendering this procedure less tempting. For given finite real- and reciprocal space cutoffs there exists an optimal α such that the accuracy of the approximated Ewald sum is the highest possible. This optimal value can be determined easily with the help of the excellent estimates for the cutoff errors derived by Kolafa and Perram[20] – essentially by demanding that the real- and reciprocal space contribution to the error should be equal. If the system

under investigation is not electrostatically neutral, the infinite sum in Eqn. (1) diverges. It can be made convergent by adding a homogeneously distributed background charge which restores neutrality – a typical situation for one-component plasma simulations. This results in an additional electroneutrality-term $E^{(n)}$ to be included in Eqn. (3), which reads (see, e.g.,[21])

$$E^{(n)} = -\frac{\pi}{2\alpha^2 V}\left(\sum_i q_i\right)^2. \tag{9}$$

Since the neutralizing background is homogeneous, the correction term in Eqn. (9) is independent of the particle positions.

The force $\mathbf{F}_i$ on particle i is obtained by differentiating the electrostatic potential energy E with respect to $\mathbf{r}_i$, i.e.,

$$\mathbf{F}_i = -\frac{\partial}{\partial \mathbf{r}_i} E. \tag{10}$$

Using Eqns. (3–8) one obtains the following Ewald formula for the forces:

$$\mathbf{F}_i = \mathbf{F}_i^{(r)} + \mathbf{F}_i^{(k)} + \mathbf{F}_i^{(d)}, \tag{11}$$

with the real space, Fourier space and dipole contributions given by:

$$\mathbf{F}_i^{(r)} = q_i \sum_j q_j \sum_{\mathbf{m}\in\mathbb{Z}^3}{}' \left(\frac{2\alpha}{\sqrt{\pi}}\exp(-\alpha^2|\mathbf{r}_{ij}+\mathbf{m}L|^2) + \frac{\operatorname{erfc}(\alpha|\mathbf{r}_{ij}+\mathbf{m}L|)}{|\mathbf{r}_{ij}+\mathbf{m}L|}\right)\frac{\mathbf{r}_{ij}+\mathbf{m}L}{|\mathbf{r}_{ij}+\mathbf{m}L|^2} \tag{12}$$

$$\mathbf{F}_i^{(k)} = q_i \sum_j q_j \frac{1}{V}\sum_{\mathbf{k}\neq\mathbf{0}} \frac{4\pi\mathbf{k}}{k^2}\exp\left(-\frac{k^2}{4\alpha^2}\right)\sin(\mathbf{k}\cdot\mathbf{r}_{ij}) \tag{13}$$

$$\mathbf{F}_i^{(d)} = -\frac{4\pi q_i}{(1+2\epsilon')V}\sum_j q_j \mathbf{r}_j. \tag{14}$$

Since the self energy in Eqn. (6) and the neutralizing contribution in Eqn. (9) are independent of particle positions, they do not contribute to the force.

2.2 The Particle Mesh Ewald Idea

The Fourier transformations involved in Eqns. (5,13) are the most time consuming part of the Ewald sum. Several methods have been proposed to address this problem, e.g., tabulation of the complete Ewald potential[22] or the use of polynomial approximations. A particularly successful approach for the latter is the expansion of the non-spherical contributions to the Ewald potential in cubic harmonics[22,23]. Apart from the difficulty of computational overhead which may strongly increase with the desired accuracy, these methods do not solve the additional problem of unfavorable scaling with particle number. At best they scale as $N^{3/2}$, which is more costly than a plain cutoff or reaction field approach.

The essential idea is *not* to avoid the Fourier transformations but to modify the problem in such a way that it permits application of the Fast Fourier Transformation (FFT, see e.g.[24] and references therein). This reduces the complexity of the reciprocal part of the

Ewald sum to $N \log N$. For increasing number of particles the real space cutoff can be varied in such a way, that this scaling applies to the complete Ewald sum. Since the FFT is a grid transformation, there are discretization problems to be solved and corresponding discretization errors to be minimized. Performing the Fourier transformations in the reciprocal space part of the Ewald sum by FFT routines is by no means straightforward:

1. The point charges with continuous coordinates have to be replaced by a grid based charge density, since the FFT is a discrete and finite transformation.

2. It is neither obvious nor true that the best grid approximation to the continuum solution of the Poisson equation is achieved by using the continuum Green function[25].

3. There are at least three possibilities for implementing the differentiation needed in Eqn. (10). They differ in accuracy and speed.

4. The procedure of assigning the forces calculated on the mesh back to the actual particles can – under certain circumstances – lead to unwanted violations of Newton's third law. They can be anything between harmless and disastrous.

The four steps involved in a particle mesh calculation are sources of various kinds of errors, originating, e.g., from discretization, interpolation or aliasing[a] problems. Since these contributions are not independent of each other (reducing one might enhance another), the only reasonable demand is the minimization of the *total* error at given computational effort.

The original literature on particle mesh routines is usually not easy to digest for the non-specialist. It is obscured by the fact that various authors approach the problem from different directions and use different notations. There exist three mesh implementations of the Ewald sum – similar in spirit but different in detail. The oldest is the original particle-particle–particle-mesh (P^3M) method of Hockney and Eastwood[25], and then there are two variants, namely, the Particle Mesh Ewald (PME) method of Darden *et al.*[26] and an extension of the latter by Essmann *et al.*[27], which is usually referred to as Smooth Particle Mesh Ewald (SPME).

In two papers by Deserno et al.[8,9] it was shown how the three methods differ in detail, and it was demonstrated that the oldest method, namely the original P^3M algorithm is actually the most accurate one, and since error estimates exist[9], this method should be the preferred mesh method. The two new ideas of the more recent methods, namely the $i\mathbf{k}$ differentiation of PME, and the analytic differentiation of the charge assignment function in real space of SPME, can both also be used in P^3M, making it the most versatile mesh algorithm on the market. Our experience shows that the break even point between a standard Ewald sum and the P^3M algorithm is around 600-800 charges, depending on implementation details and desired accuracy. The algorithm can also be efficiently parallelized[28,29].

2.3 P^3M in a Nutshell

The P^3M method maps the system onto a mesh, such that the necessary Fourier transformations can be accomplished by Fast Fourier routines. At the same time the simple Coulomb

[a] A finite grid cannot represent arbitrarily large $\mathbf{k}$-vectors. Instead, they are folded back into the first "Brillouin zone" and distort there the true spectrum. This effect is usually referred to as "aliasing". See, e.g., Sec. 12.1 in Ref.[24].

Green function $4\pi/k^2$ is adjusted by a functional minimization of an error functional as to make the result of the mesh calculation most closely resemble the continuum solution.

The first step, i.e., generating the mesh based charge density ρ_{M} (defined at the mesh points $\mathbf{r}_p$), is carried out with the help of a charge assignment function W:

$$\rho_{\mathrm{M}}(\mathbf{r}_p) = \frac{1}{h^3} \sum_{i=1}^{N} q_i W(\mathbf{r}_p - \mathbf{r}_i) \tag{15}$$

Here h is the mesh spacing, and the number of mesh points $N_{\mathrm{M}} = L/h$ along each direction should preferably be a power of two, since in this case the FFT is most efficient. The charge assignment function is classified according to its order P, i.e. between how many grid points – per coordinate direction – each charge is distributed. For W a cardinal B-spline[30] is chosen, which is a piecewise polynomial function of weight one. The order P gives the number of sections in the function. The first 7 cardinal-B-splines are sketched in Fig. 1. Its Fourier transform is

$$\tilde{W}(\mathbf{k}) = h^3 \left(\frac{\sin(\frac{1}{2}k_x h)}{\frac{1}{2}k_x h} \frac{\sin(\frac{1}{2}k_y h)}{\frac{1}{2}k_y h} \frac{\sin(\frac{1}{2}k_z h)}{\frac{1}{2}k_z h} \right)^P \tag{16}$$

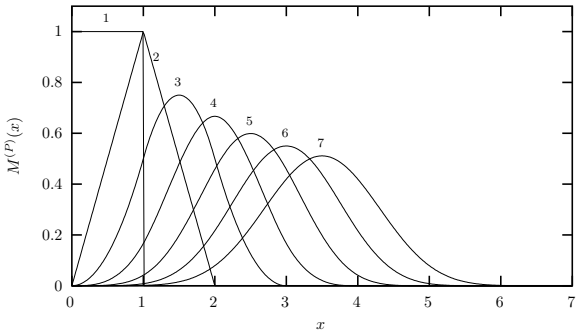


Figure 1. Sketch of the first 7 cardinal-B-splines $M^{(P)}(x)$, parameterized by P. Note that the charge assignment functions $W^{(P)}(x)$ for the P^{3}M algorithm are just the "centered" B-splines.

In a second step the mesh based electric field $\mathbf{E}(\mathbf{r}_p)$ is calculated. Basically, the electric field is the derivative of the electrostatic potential, but there exist several alternatives for implementing the differentiation on a lattice[8]. Here we will restrict ourselves to the case of $i\mathbf{k}$-*differentiation*, which works by multiplying the Fourier transformed potential with $i\mathbf{k}$. In this case $\mathbf{E}(\mathbf{r}_p)$ can be written as

$$\mathbf{E}(\mathbf{r}_p) \;=\; \overleftarrow{\mathrm{FFT}} \left[-i\mathbf{k} \times \overrightarrow{\mathrm{FFT}} \left[\rho_{\mathrm{M}}\right] \times \hat{G}_{\mathrm{opt}} \right] (\mathbf{r}_p) \tag{17}$$

In words, $\mathbf{E}(\mathbf{r}_p)$ is the *backward* finite Fourier transform of the product of $-i\mathbf{k}$, the *forward* finite Fourier transform of the mesh based charge density ρ_{M} and the so-called optimal

influence function $\hat{G}_{\text{opt}}$, given by

$$\hat{G}_{\text{opt}}(\mathbf{k}) = \frac{\tilde{\boldsymbol{D}}(\mathbf{k}) \cdot \sum_{\mathbf{m}\in\mathbb{Z}^3} \tilde{U}^2(\mathbf{k}+\frac{2\pi}{h}\mathbf{m})\tilde{\boldsymbol{R}}(\mathbf{k}+\frac{2\pi}{h}\mathbf{m})}{|\tilde{\boldsymbol{D}}(\mathbf{k})|^2 \left[\sum_{\mathbf{m}\in\mathbb{Z}^3} \tilde{U}^2(\mathbf{k}+\frac{2\pi}{h}\mathbf{m})\right]^2} \tag{18}$$

$$\text{with} \qquad \tilde{\boldsymbol{R}}(\mathbf{k}) := -i\mathbf{k}\frac{4\pi}{k^2}e^{-k^2/4\alpha^2} \tag{19}$$

$$\text{and} \qquad \tilde{U}(\mathbf{k}) := \tilde{W}(\mathbf{k})/h^3 \tag{20}$$

Here $\tilde{\boldsymbol{D}}(\mathbf{k})$ is the Fourier transform of the employed differentiation operator, which is simply $i\mathbf{k}$ in our case. Finally, one arrives at the force on particle i, i.e. the replacement of Eqn. (13):

$$\mathbf{F}_i = q_i \sum_{\mathbf{r}_p\in\mathbb{M}} \mathbf{E}(\mathbf{r}_p)W(\mathbf{r}_i - \mathbf{r}_p) \tag{21}$$

Hereby the sum extends over the complete mesh $\mathbb{M}$.

Although the presented formulas (15–21) look somewhat complicated, it is rather easy to implement them step by step. If the real space cutoff r_{max} is chosen small enough (so that the real space contribution (12) can be calculated in order N), the complete algorithm is essentially of order $N \log N$, see the article by Petersen[31] for a nice exposition of this point.

2.4 How and Why to Control Errors

An investigation of the errors connected with particle mesh Ewald methods is important for three reasons:

1. The procedure of discretization introduces new sources of errors in addition to the ones originating from real and reciprocal space cutoffs.

2. Comparing the efficiency of different mesh methods is only fair if it is done at the same level of accuracy.

3. The tuning parameters should be chosen in such a way as to run the algorithm at its optimal operation point.

The two last points of course apply to any numerical method to compute energy or forces for the Coulomb (or dipolar) sum. Errors can tell us if we might see artifacts in simulations due to a poorly converged sum. They can tell us how the algorithm scales at its optimal point, and they can help us save a lot of expensive computer time. Unfortunately some of the free or commercially available computer programs choose parameter combinations automatically according to some more or less unknown rules, hidden in the program, which is a very dangerous route since after all the user needs to interprete the data. Therefore we stress here the point that for all our implemented routines we have error estimates, and we *use them*.

However, there is no unique or optimal measure of accuracy. If molecular dynamics (MD) simulations are performed, the main interest lies in errors connected with the force,

while in Monte-Carlo (MC) simulations one is concerned with accurate energies. In the simulation of ensemble averages it is the global accuracy, measured e.g. by root mean square quantities, which is important. In the simulation of rare events local accuracy and maximal errors are relevant as well. Errors in the force can be due to their magnitude or due to their direction. And finally, one might be interested in absolute or relative errors.

Whatever quantity one decides to look at, it can be investigated as a function of system parameters like particle separation or distribution, *tuning* parameters like α, mesh size or interpolation order and *components* of the algorithm, e.g., interpolation or differentiation scheme or splitting function $f(r)$. This obviously gives rise to a very large number of combinations. In other words: The corresponding parameter space is large and nontrivial, i.e., general statements concerning the performance of one method can usually not be extracted from low-dimensional cuts through this space, because different methods scale differently with respect to their parameters.

We are concerned with one measure of accuracy, namely the root mean square (rms) error in the force given by

$$\Delta F := \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\mathbf{F}_i - \mathbf{F}_i^{\mathrm{exa}}\right)^2}, \tag{22}$$

where $\mathbf{F}_i$ is the force on particle number i calculated via some mesh method and $\mathbf{F}_i^{\mathrm{exa}}$ is the *exact* physical force on that particle, calculable e.g. by a well converged standard Ewald sum. There exist error estimates for the real space and Fourier space contribution to this error for the standard Ewald sum[20], for the PME method[31], and for the P^{3}M method[9].

Here we show the dependence of the root mean square error in the force in Eqn. (22) on the number of charged particles and their valence.[b]. Since the assumptions and arguments involved are of a rather general nature, the result is not specific to a certain type of Ewald method.

It is reasonable to assume that the error in the force on particle i can be written as

$$\Delta\mathbf{F}_i \;:= \mathbf{F}_i - \mathbf{F}_i^{\mathrm{exa}} \;=\; q_i \sum_{j\neq i} q_j\, \boldsymbol{\chi}_{ij}. \tag{23}$$

The idea behind this Ansatz is that – just as it is true for $\mathbf{F}_i$ – the *error* in $\mathbf{F}_i$ originates from the $N-1$ interactions of particle i with the other charged particles, and each contribution should be proportional to the product of the two charges involved. The vector $\boldsymbol{\chi}_{ij}$ gives the direction and magnitude of this error for two unit charges and depends on their separation and orientation as well as on the specific algorithm used for calculating the electrostatic forces. For this term it is further assumed that

$$\left\langle \boldsymbol{\chi}_{ij}\cdot\boldsymbol{\chi}_{ik}\right\rangle \;=\; \delta_{jk}\left\langle \boldsymbol{\chi}_{ij}^2\right\rangle \;=:\; \delta_{jk}\,\chi^2, \tag{24}$$

where averaging over the particle configurations is denoted by the angular brackets $\langle\cdots\rangle$. The underlying assumption that contributions from different particles are uncorrelated is certainly not always true, think, e.g., of highly ordered or strongly inhomogeneous particle distributions. However, it is a sensible one for *random* systems. Obviously, the term $\langle\boldsymbol{\chi}_{ij}^2\rangle$

[b] Note that this is by no means the only interesting measure of accuracy

– the mean square force error for two unit charges – can no longer depend on i and j and is thus written as χ^2. Using Eqns. (23,24), it follows

$$\left\langle (\Delta \mathbf{F}_i)^2 \right\rangle \;=\; q_i^2 \sum_{j\neq i} \sum_{k\neq i} q_j q_k \langle \boldsymbol{\chi}_{ij} \cdot \boldsymbol{\chi}_{ik} \rangle \;\approx\; q_i^2 \, \chi^2 \, \mathcal{Q}^2, \tag{25}$$

where the important quantity $\mathcal{Q}^2$ is defined as

$$\mathcal{Q}^2 \;:=\; \sum_{j=1}^{N} q_j^2. \tag{26}$$

We further assume that

$$\left\langle \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\Delta \mathbf{F}_i)^2} \right\rangle \approx \sqrt{\frac{1}{N} \sum_{i=1}^{N} \langle (\Delta \mathbf{F}_i)^2 \rangle}, \tag{27}$$

which can be shown true for reasonably large systems by the law of large numbers along the line of reasoning of Ref.[9] to end up with the final relation

$$\Delta F \;\;\approx\;\; \chi \, \frac{\mathcal{Q}^2}{\sqrt{N}}. \tag{28}$$

Thus, the scaling of the rms error in the force with particle number and valence is given by the factor $\mathcal{Q}^2 N^{-1/2}$, whereas the prefactor χ – which cannot be obtained by such simple arguments – contains the details of the method and is independent of the simulated system. Note that any information on the valence distribution enters only through the value of $\mathcal{Q}^2$.

The most interesting ingredient of the P^{3}M method is the optimal influence function from Eqn. (18). It is constructed such that the result of the mesh calculation is as close as possible to the solution of the original continuum problem. More precisely, the P^{3}M method is derived from the requirement that the resulting Fourier space contribution to the force minimizes the the following error measure Q:

$$Q := \frac{1}{h^3} \int_{h^3} \mathrm{d}^3 r_1 \int_{L^3} \mathrm{d}^3 r \left[\mathbf{F}(\mathbf{r}; \mathbf{r}_1) - \boldsymbol{R}(\mathbf{r}) \right]^2 \tag{29}$$

$\mathbf{F}(\mathbf{r}; \mathbf{r}_1)$ is the Fourier space contribution of the force between two unit charges at positions $\mathbf{r}_1$ and $\mathbf{r}_1 + \mathbf{r}$ as calculated by the P^{3}M method (note that due to broken rotational and translational symmetry this does in fact depend on the coordinates of *both* particles), and $\boldsymbol{R}(\mathbf{r})$ is the corresponding exact reference force (whose Fourier transform is just Eqn. (19)). The inner integral over $\mathbf{r}$ scans all particle separations, whereas the outer integral over $\mathbf{r}_1$ averages over all possible locations of the first particle within a mesh cell. Obviously, up to a factor L^{-3} this expression is just the mean square error in the force for two unit charges, in other words, the quantity χ^2 from Eqn. (24). This provides a link between the rms error of an N particle system and the error Q from Hockney and Eastwood: Using Eqn. (28) one obtains

$$\Delta F \;\approx\; \mathcal{Q}^2 \sqrt{\frac{Q}{N L^3}}. \tag{30}$$

It is important to realize that Hockney and Eastwood not only provide a closed expression for the optimal influence function $\hat{G}_{\rm opt}$, but also a closed expression for the corresponding "optimal error" $Q_{\rm opt} = Q[\hat{G}_{\rm opt}]$:

$$Q_{\rm opt} = \frac{1}{L^3} \sum_{\mathbf{k}\in\hat{\mathbb{M}}} \Bigg\{ \sum_{\mathbf{m}\in\mathbb{Z}^3} \Big|\tilde{\boldsymbol{R}}(\mathbf{k}+\frac{2\pi}{h}\mathbf{m})\Big|^2 - \frac{\Big|\tilde{\boldsymbol{D}}(\mathbf{k})\cdot\sum_{\mathbf{m}\in\mathbb{Z}^3}\tilde{U}^2(\mathbf{k}+\frac{2\pi}{h}\mathbf{m})\tilde{\boldsymbol{R}}^*(\mathbf{k}+\frac{2\pi}{h}\mathbf{m})\Big|^2}{|\tilde{\boldsymbol{D}}(\mathbf{k})|^2\left[\sum_{\mathbf{m}\in\mathbb{Z}^3}\tilde{U}^2(\mathbf{k}+\frac{2\pi}{h}\mathbf{m})\right]^2} \Bigg\}. \tag{31}$$

The outer sum extends over all $\mathbf{k}$-vectors of the Fourier transformed mesh $\hat{\mathbb{M}}$, and the asterisk denotes the complex conjugate. Once again, in the special case of i$\mathbf{k}$-differentiation one has $\tilde{\boldsymbol{D}}(\mathbf{k}) = \mathrm{i}\mathbf{k}$.

Admittedly, Eqn. (31) looks rather complicated. Still, in combination with Eqn. (30) it gives the rms force error of the P^{3}M method or – more precisely – of its Fourier space contribution. After all, the computation of $Q_{\rm opt}$ and that of $\hat{G}_{\rm opt}$ are quite similar. It should be emphasized that the formula (31) for the optimal Q-value, just like the one for the optimal influence function in Eqn. (18), is of a very general nature. It does also work for different charge assignment functions, reference forces or any differentiation scheme which can be expressed by an operator $\tilde{\boldsymbol{D}}(\mathbf{k})$.

The corresponding rms error in the force from the real space contribution in Eqn (12) has been derived by Kolafa and Perram[20] and is provided here for reference purpose:

$$\Delta F^{(r)} \approx \frac{2\mathcal{Q}^2}{\sqrt{N r_{\rm max} L^3}}\, \exp(-\alpha^2 r_{\rm max}^2). \tag{32}$$

With these two estimates at hand it is easy to determine the optimal value of the splitting parameter α via a stand-alone program, which takes the relevant system parameters (N, $\mathcal{Q}^2$, L) and specifications of the algorithm ($r_{\rm max}$, $N_{\rm M}$, P) as its input, where $N_{\rm M}$ denotes the number of mesh points. If real and reciprocal space contribution to the error, $\Delta F^{(r)}$ and $\Delta F^{(k)}$ respectively, are assumed to be statistically independent, the total error is given by

$$\Delta F = \sqrt{\left(\Delta F^{(r)}\right)^2 + \left(\Delta F^{(k)}\right)^2}. \tag{33}$$

This quantity has to be minimized with respect to α. In most cases it is, however, accurate enough to use the following approximation: Determine the value of α at which the real and reciprocal space contribution to the rms force error are equal.

2.5 Alternatives to Ewald

Another way of computing Eq.(1) is via a convergence factor

$$E = \lim_{\beta\to 0} \frac{1}{2} \sum_{\mathbf{n}}{}' \sum_{ij} \frac{q_i q_j \exp\left(-\beta|\mathbf{r}_{ij}+\mathbf{n}L|\right)}{|\mathbf{r}_{ij}+\mathbf{n}L|}. \tag{34}$$

This approach is used in the Lekner[32] and Sperb[33,34] methods to efficiently sum up the 3D Coulomb sum. Although the method in its original versions has $\mathcal{O}(N^2)$ complexity,

Strebel and Sperb have developed a factorization approach which yields an $\mathcal{O}(N \log N)$ algorithm[35].

Other advanced methods of $\mathcal{O}(N \log N)$ are tree algorithms[36], which are the first order approximation of even better, so-called fast multipole methods[37] (FMM). These can reach a linear complexity, but at the expense of a heavy computational overhead which makes these methods advantageous only for a very large number of charges ($N \approx 100\,000$)[38], see also the contribution of Grubmüller in this volume. Another recent approach is to use multigrid methods which solve the Poisson equation in real space[39,40]. These have the advantage that they should be more useful on massively parallel architectures since an efficient implementation of 3D FFTs is difficult on those machines. Another advantage could be the fact that for MD simulations the previous Poisson solution can still be used as a starting point for the multigrid iteration thereby improving drastically the convergence speed.

All these alternative methods (except for the Strebel algorithm) share the property that in principle they can be applied also to geometrical situations where only one or two dimensions are periodically replicated. The Ewald like algorithms can be formulated for those geometries as well, however they loose tremendously in speed. However, several tricks can be applied to accelerate them.

3 Two Periodic Dimensions

For thin polyelectrolytes films, interactions of charged species with membranes, or in general geometries, where only two dimensions are periodically replicated, and the third one has a finite extend h ($2D + h$ geometry), the original Ewald sum has to be modified. For such systems Ewald based formulas are only slowly convergent, have mostly $\mathcal{O}(N^2)$ scalings and no "a priori" error estimates exist[41]. A fast Ewald based method has been put forward in[42], where the Fourier sum is specially treated with an integral approximation. However, in our opinion, this introduces non-controllable errors, which makes the method useless[43]. Also the Lekner method, which is based on a resummation of the force sum, can be used, however it has an $\mathcal{O}(N^2)$ scaling, and special care has to be taken for particles which are nearby in the non-periodic dimension, see also the discussion of Mazars[44] on accuracy problems. Recently we proposed a new method called **MMM2D**[11,45] which has an $\mathcal{O}(N^{5/3})$ complexity and full error control. It is based on a convergence factor approach similar to **MMM**[33]. In two dimensions the convergence factor based methods and the Ewald sum methods yield exactly the same results. However, this will still only allow simulations including up to a few thousand charges. There have been early attempts to use the full 3D Ewald sum also for these slab problems. The main idea is to fill only parts of the simulation box with charges and to leave some space empty, in an attempt to decouple the interactions in the third dimension[46–48]. Since each image layer is globally neutral, one hopes that their interactions decay as they become more and more distant, i.e. as the size of the gap is increased. In this way one could make use of any advanced 3D Ewald implementation, see also Ref.[49] for a variant of this idea.

3.1 The ELC Method

Up to recently the 3D methods using a gap had to be checked on a trial and error basis. We recently greatly improved this situation by deriving a term, called electrostatic layer correction (**ELC**), which subtracts the interactions due to the unwanted layers[12,13]. In this way the necessary gap length can be largely reduced. The combination of that term with any three dimensional summation method with slab–wise summation order will yield the exact electrostatic energy. Since the change in the summation order is done by adding a very simple term, any three dimensional summation method with the standard spherical summation order can be used. The **ELC**-term can be evaluated easily in a time linear in the number of charges, hence the whole method scales like the underlying standard summation method. We also developed an error formula for the maximal pairwise error in the energy and forces of the layer correction term, hence the precision of this method can be tuned to any desired value, when used in conjunction with other error estimates for the standard summation method[20,9]. Using, for example, the P^{3}M method one obtains an $N \log N$ scaling with well controlled errors also for the $2D + h$ geometry, which up to now seems to be the optimal choice.

We just try to give here some insight into the basic idea of changing the summation order. Consider this time a system of N particles with charges q_i and positions $\mathbf{r}_i = (x_i, y_i, z_i)$ that reside in a box of edges $L \times L \times h$, where $h = \max_{i,j} |z_i - z_j|$ is the maximal z–distance of two particles. The basic idea is to expand this slab system in the non-periodic z–coordinate to a system with periodicity in all three dimensions. More precisely, the original box of size $L \times L \times h$ is placed inside a box of size $L \times L \times L_z$ where $L_z >> h$ sufficiently large. Then this box is replicated periodically in all three dimensions. The result is a three-dimensional periodic system with empty space regions ("gaps") of height $\delta := L_z - h$ (see Fig. 2). δ will be called gap size in the following.

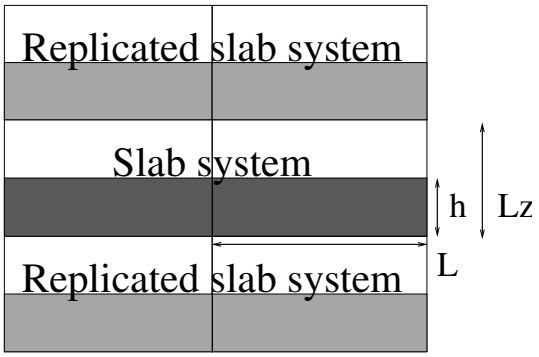


Figure 2. Schematic representation of a fully periodically replicated slab system

Since the electrostatic potential is only finite if the total system is charge neutral, the additional image layers (those layers above or below the original slab system) are charge neutral, too. Now let us consider the n^{th} image layer which has an offset of nL_z to the original layer. If nL_z is large enough, each particle of charge q_j at position $(x_j, y_j, z_j + nL_z)$ and its replicas in the x, y-plane can be viewed as constituting a homogeneous charged sheet

of charge density $\sigma_j = \frac{q_j}{L^2}$. The potential of such a charged sheet at distance z is $2\pi\sigma_j|z|$. Now we consider the contribution from a pair of image layers located at $\pm nL_z$, $n > 0$ to the energy of a charge q_i at position (x_i, y_i, z_i) in the central layer. Since $|z_j - z_i| < nL_z$, we have $|z_j - z_i + nL_z| = nL_z + z_j - z_i$ and $|z_j - z_i - nL_z| = nL_z - z_j + z_i$, and hence the interaction energy from those two image layers with the charge q_i vanishes by charge neutrality:

$$2\pi q_i \sum_{j=1}^{N} \sigma_j(|z_j - z_i + nL_z| + |z_j - z_i - nL_z|) = 4\pi q_i nL_z \sum_{j=1}^{N} \sigma_j = 0. \tag{35}$$

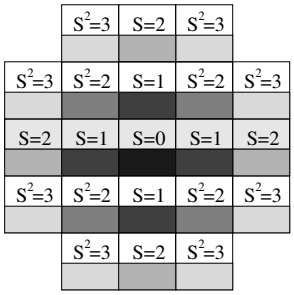
The only errors occurring are those coming from the approximation of assuming homogeneously charged, infinite sheets instead of discrete charges. This assumption should become better when increasing the distance nL_z from the central layer.

However, in a naive implementation, even large gap sizes will result in large errors[48]. This is due to the order of summation for the three dimensional Coulomb sum, which is spherical by convention. This order implies that with increasing shell cutoff S the number of image shells grows faster than the number of shells of the primary layer, namely $\mathcal{O}(S^3)$ versus $\mathcal{O}(S^2)$ (see Fig. 3(a)). In other words, we include the unwanted terms faster than the actually wanted terms. Also the image layers are not really infinite charged sheets but are truncated due to the cutoff. Yeh and Berkowitz[48] already suggested that this problem can be solved by changing the order of summation. Smith has shown that by adding to the Coulomb energy the term

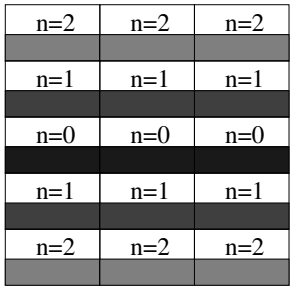
$$E_c = 2\pi M_z^2 - \frac{2\pi \mathbf{M}^2}{3}, \tag{36}$$

where $\mathbf{M} = \sum q_i \mathbf{r}_i$ is the total dipole moment, one obtains the result of a slab–wise summation instead of the spherical limit[50]. Slab–wise summation refers to the sum $\sum_{|n|\geq 0} E_l(n)$, where $E_l(n)$ denotes the energy, calculated in circular summation order, resulting from the image layer with shift nL_z in the z–coordinate. Technically this is the order where we first treat the original layer and then add the image layers grouped in symmetrical pairs (see Fig. 3(b)). Obviously this summation order fits much better to the charged sheet argumentation given above. Although this is a major change in the summation order, the difference given by Eq. (36) is a very simple term. In fact, Smith shows that changes of the summation order always result in a difference that depends only on the total dipole moment.

Applying this slab–wise summation order, Yeh and Berkowitz showed that a gap size of at least h is normally sufficient to obtain an moderately accurate result. Therefore the result of a standard three dimensional summation method plus the shape–dependent term given by Eq. (36), which we refer to as a *slab–wise method*, can be used to obtain a good approximation to the result for the slab geometry with the same computational effort as for the underlying three dimensional summation method (no matter if a simple or sophisticated method is used). One drawback is that no theoretical estimates exist for the error introduced by the image layers. Therefore one might be forced to use even larger gaps to assure that no artifacts are produced by the image layers. One deducible artifact is that the pairwise error will be position dependant. Particles in the middle of the slab will see no effect of the image layers due to symmetry, and particles near the surface will encounter for the same reason the largest errors, which is definitely an unwanted feature for studying surface



(a) Schematic view of the spherical summation order. S is the length of the box offset.



(b) Schematic view of the slab–wise summation order. n is the z offset of the box, the spherical summation order in the x, y–plane is not shown.

Figure 3. Schematic views of different summation orders.

effects. Therefore averaging error measures like the commonly used RMS force error should not be applied without additional checks for the particles near the surfaces.

The other drawback is that normally the box now will have a significantly larger L_z/L. But at least for Ewald type methods the computation time is proportional to this fraction. This is easy to see as the number of k–space vectors in the z direction must be proportional to L_z to maintain a fixed resolution and therefore error. It is verified experimentally that a gap of at least h is needed. For a cubic system $h = L$ therefore the computation time at least doubles.

Nevertheless because of the bad scaling of the known methods for slab geometries like the one by Parry[51,52] ($\mathcal{O}(N^2)$) or **MMM2D**[11,45] ($\mathcal{O}(N^{5/3})$), for particle numbers above $N \approx 1000$ using slab–wise methods is a great improvement. Fortunately one can do even better!

We briefly show how the **ELC** term can be computed; consult the original article for more details[12,13]. The method relies on the far formula of **MMM2D**[11,45]. For the following analysis there is no special restriction on h except for $h < L_z$, which is true even if the

$L \times L \times L_z$–box is completely filled.

We start with a formal definition of the Coulomb energy of the slab system

$$E = \frac{1}{2} \sum_{S=0}^{\infty} \sum_{\substack{\mathbf{n} \in \mathbb{Z}^2 \times \{0\} \\ n_x^2 + n_y^2 = S}} \sum_{i,j=1}^{N}{}' \frac{q_i q_j}{|\mathbf{r}_i - \mathbf{r}_j + \Lambda \mathbf{n}|} \,. \tag{37}$$

$\Lambda = \mathrm{diag}(L, L, L_z)$ is a diagonal matrix describing the shape of the box. The image boxes are denoted with the vector $\mathbf{n} = (n_x, n_y, n_z)$, where $n_z = 0$ for now. The prime on the inner summation indicates the omission of the self–interaction $i = j$ in the primary box $\mathbf{n} = (0, 0, 0)$ (i. e. the singular case). For the surrounding dielectric medium we assume vacuum boundary conditions.

We now expand the system to a fully three-dimensional periodic system, where L_z determines the period in the z-coordinate as in the previous section. We can rewrite the energy as

$$E = E_s + E_c + E_{lc} \,, \tag{38}$$

where

$$E_s = \frac{1}{2} \sum_{S=0}^{\infty} \sum_{\substack{\mathbf{n} \in \mathbb{Z}^3 \\ \mathbf{n}^2 = S}} \sum_{i,j=1}^{N}{}' \frac{q_i q_j}{|\mathbf{r}_i - \mathbf{r}_j + \Lambda \mathbf{n}|} \,. \tag{39}$$

denotes the standard three-dimensional Coulomb–sum with spherical limit. To evaluate this expression one can use any of the efficient algorithms, starting with the classical Ewald summation up to modern methods like fast multipole methods[37] or mesh based algorithms[8]. E_c again denotes the shape–dependent term given by Eq. (36) and finally

$$E_{lc} = -\frac{1}{2} \sum_{\substack{T \in \mathbb{Z} \\ T > 0}} \sum_{n_z = \pm T} \sum_{S=0}^{\infty} \sum_{\substack{\mathbf{n} \in \mathbb{Z}^2 \times \{n_z\} \\ \mathbf{n}^2 = S}} \sum_{i,j=1}^{N} \frac{q_i q_j}{|\mathbf{r}_i - \mathbf{r}_j + \Lambda \mathbf{n}|} \,. \tag{40}$$

denotes the contribution of the image layers, for which we are going to derive a new expression in the following.

We start with the expression for the energy induced by an image layer at z–offset $n_z \neq 0$:

$$E_l(n_z) = -\frac{1}{2} \sum_{S=0}^{\infty} \sum_{\substack{\mathbf{n} \in \mathbb{Z}^2 \times \{n_z\} \\ \mathbf{n}^2 = S}} \sum_{i,j=1}^{N} \frac{q_i q_j}{|\mathbf{r}_i - \mathbf{r}_j + \Lambda \mathbf{n}|} \,. \tag{41}$$

It can be shown rigorously, although this is non–trivial, that

$$E_l(n_z) = -\frac{1}{2} \lim_{\beta \to 0} \sum_{\mathbf{n} \in \mathbb{Z}^2 \times \{n_z\}} \sum_{i,j=1}^{N} \frac{q_i q_j e^{-\beta |\mathbf{r}_i - \mathbf{r}_j + \Lambda \mathbf{n}|}}{|\mathbf{r}_i - \mathbf{r}_j + \Lambda \mathbf{n}|} \,. \tag{42}$$

This is a convergence factor approach with a convergence factor of $e^{-\beta |\mathbf{r}_i - \mathbf{r}_j + \Lambda \mathbf{n}|}$. Note that this approach is exact only for two-dimensional systems, for three-dimensional systems Eqs. (41) and (42) differ by a multiple of the dipole moment[5,6].

In[53,11] one can find a proof for this equation and an efficient way of calculating E_l for charge neutral systems. We do not want to go through the full derivation again; it consists of the application of Poisson's summation formula along both periodic coordinates and performing the limit $\beta \to 0$ analytically. One obtains

$$E_{lc}(n_z) = -\frac{1}{2} \sum_{i,j=1}^{N} q_i q_j \phi(\mathbf{r}_i - \mathbf{r}_j + \Lambda \mathbf{n}) \,, \tag{43}$$

where ϕ is an artificial pairwise potential that yields the total Coulomb energy and its derivative produces the pairwise forces for the periodic system, and $k_\parallel = (k_x, k_y)$ is a Fourier variable with integer values.

For now we only have a formula for the contribution of one image layer, so we still have to sum over all n_z. This task can be performed analytically. The terms $2\pi|z|/L^2$ can be omitted since they are exactly the homogeneous sheet potential and we have seen before that this cancels out for charge neutral systems (see Eq. (35)).

The summation over n_z of the remaining sums over p and q is fairly easy to perform using the geometric series (as these sums are absolutely convergent, exchanging the summation over n_z and the summations over (k_x, k_y) is possible). Combining the terms for $\pm n_z$ again we obtain

$$E_{lc} = \sum_{i,j=1}^{N} q_i q_j \psi(\mathbf{r}_i - \mathbf{r}_j + \Lambda \mathbf{n}) \,, \tag{44}$$

where

$$\begin{aligned} \psi(x,y,z) = & \frac{4}{L} \sum_{k_x,k_y>0} \frac{\cosh(2\pi k_\parallel z_{ij}/L)}{k_\parallel (e^{2\pi k_\parallel L_z/L} - 1)} \cos(2\pi k_x x_{ij}/L) \cos(2\pi k_y y_{ij}/L) + \\ & \frac{2}{L} \sum_{k_x>0} \frac{\cosh(2\pi k_x z_{ij}/L) \cos(2\pi k_x x_{ij}/L)}{k_x (e^{2\pi k_x L_z/L} - 1)} + \\ & \frac{2}{L} \sum_{k_y>0} \frac{\cosh(2\pi k_y z_{ij}/L) \cos(2\pi k_y y_{ij}/L)}{k_y (e^{2\pi k_y L_z/L} - 1)} \,. \end{aligned} \tag{45}$$

The forces can be obtained from that by simple differentiation since the sums are absolutely convergent. Although the form in Eq.(45) has a much better convergence than the original form in Eq.(40), its main advantage is a linear computation time with respect to the number of particles N. To see this, the equation has to be rewritten using the addition theorems for the cosine and the hyperbolic cosine. For each $k_\parallel$ one first calculates the

sixteen terms

$$\begin{aligned}
\chi_{(c/s,c/s,c/s)} &= \sum_{i=1}^{N} q_i \cosh / \sinh(2\pi k_\parallel z_i/L) \cos / \sin(2\pi k_x x_i/L) \cos / \sin(2\pi k_y y_i/L)\,, \\
\chi_{(x,c/s,c/s)} &= \sum_{i=1}^{N} q_i \cosh / \sinh(2\pi k_x z_i/L) \cos / \sin(2\pi k_x x_i/L)\,, \\
\chi_{(y,c/s,c/s)} &= \sum_{i=1}^{N} q_i \cosh / \sinh(2\pi k_y z_i/L) \cos / \sin(2\pi k_y y_i/L)\,,
\end{aligned} \tag{46}$$

where the indices in the obvious way determine which of the functions cosine (hyperbolicus) or sinus (hyperbolicus) are used. Then we evaluate

$$\begin{aligned}
E_{lc} =& \frac{4}{L} \sum_{k_x,k_y>0} \frac{1}{(e^{2\pi k_\parallel L_z/L}-1)k_\parallel} \Big(\chi^2_{(ccc)} + \chi^2_{(csc)} + \chi^2_{(ccs)} + \chi^2_{(css)} - \\
& \qquad \chi^2_{(scc)} - \chi^2_{(ssc)} - \chi^2_{(scs)} - \chi^2_{(sss)}\Big) + \\
& \frac{2}{L} \sum_{k_x>0} \frac{1}{(e^{2\pi k_x L_z/L}-1)k_x} \Big(\chi^2_{(xcc)} + \chi^2_{(xcs)} - \chi^2_{(xsc)} - \chi^2_{(xss)}\Big) + \\
& \frac{2}{L} \sum_{k_y>0} \frac{1}{(e^{2\pi k_y L_z/L}-1)k_y} \Big(\chi^2_{(ycc)} + \chi^2_{(ycs)} - \chi^2_{(ysc)} - \chi^2_{(yss)}\Big)
\end{aligned} \tag{47}$$

Similar expansions using the same sixteen terms can also be found for the forces. Obviously this has linear computation time with respect to the number of particles, as the only summations over all the particles occur in the χ_*. The infinite summation over $k_\parallel$ can be cutoff at some value to achieve the desired accuracy, using the error estimates presented by Arnold et al[12].

To summarize, the main profits of the **ELC** term are, that it scales as the number N of particles and that it has a rigorous error bound. Moreover this error bound can be used to estimate the size of the image layer contribution and therefore gives a bound on the error introduced by slab–wise methods as proposed by Yeh and Berkowitz[48]. We found that the error for these methods decays exponentially in L_z/L. However, the errors are not uniformly distributed over the slab, namely they are worst at the surfaces of the slabs, hence the maximal pairwise error should be used instead of the usually RMS-errors of (22). In the second paper on **ELC**[13] we considered in detail the implementation of the layer correction for the standard Ewald and the P^{3}M method. There we also derived anisotropic Ewald error formulas, gave some fundamental guidelines for optimization, demonstrated the accuracy, and gave error formulas and computation times for typical systems.

4 The Coulomb Sum in One Periodic Dimension

Boundary conditions where only one dimension is periodic, and the other two are open or finite appear, for example, in physical situations such as electronic structures of supported

structures on metal surfaces, such as steps and atomic chains in one dimensions, or in systems which can model a charged stiff polymer or DNA piece, where for avoiding end effects the rod is made infinitely long[54–56]. It was only very recently that a 1D Ewald method (EW1D) for these systems has been developed[57,58], although also the Lekner method[32] developed earlier can be used, see however[44]. Again, both are $\mathcal{O}(N^2)$ methods which converge rather slow, and no error estimates have been provided.

We recently developed a method which we called **MMM1D**[59], which is basically the **MMM2D** method adapted to 1D. Again, rigorous error bounds can be derived, and although the method is $\mathcal{O}(N^2)$, it is much faster than EW1D, and easier to use. We applied this method already to charged rod systems with good results[55,56].

As mentioned in (2.5) in principle the FMM, multigrid methods, or tree codes can handle this situation, but they are quite involved for the normally only small number of charges involved, and error estimates are not easy to obtain.

Also a modified Ewald method in which the summation of the reciprocal-scape vectors was modified[60], similar to the one used by Kawata and Mikami[61] exists, but again the approximations made seem hard to control which render the method rather useless.

And finally, one can in principle use any 3D Ewald method and separate the rod images sufficiently to ensure that the interactions are negligible, which can be ensured for systems where mobile counterions are present which can screen the interactions. This has been applied often by Deserno et al.[10,62,63]. As a side remark, similar to the **ELC** method, by changing the summation order and using the prescription of Smith[50], a corrected 3DEW summation should also be possible, but it might again not be easy to solve the problem of how to control the error.

5 The Dipolar Ewald Summation in 3D

In the previous section we treated methods to deal with the Coulomb, or more generally, interactions which vary with $1/r$. However, there are also systems of interest, which can be modeled by interacting point dipoles. Substances of that kind are ferrofluids, which are basically dispersed magnetic particles[64], or magneto-rheological (MR) or electro-rheological (ER) fluids, or solvents which can be modeled approximately by dipolar interactions like water. For periodic geometries the Ewald method is again applicable. In the simplest implementations the involved computation time grows like N^2, or at best like $N^{3/2}$, if the cutoff is optimally varied with the splitting parameter[7]. For MC simulations, knowing the energy formulas is sufficient, whereas for MD simulations, we need to know forces and torques. In this part we give a reliable error estimate for the energy, for the forces, and for the torques, when computed via the standard Ewald sum. Moreover we will give a detailed discussion on the optimization of the parameters, which will lead to the most efficient parameters for a predefined error in each observable quantity. This can all be done prior to the actual simulation, ensuring thus optimal performance at optimally controlled errors. The exposition is heavily based on the paper by Wang and Holm[65].

Consider a system of N particles with a point-dipole $\boldsymbol{\mu}_i$ at their center position $\mathbf{r}_i$ in a cubic simulation box of length L. If periodic boundary conditions are applied, the total

electrostatic energy of the box is given by

$$U = \frac{1}{2} \sum_{\mathbf{n}\in\mathbb{Z}^3}{}' \sum_{i,j=1}^{N} \left\{ \frac{\boldsymbol{\mu}_i \cdot \boldsymbol{\mu}_j}{\mid \mathbf{r}_{ij} + \mathbf{n} \mid^3} - \frac{3[\boldsymbol{\mu}_i \cdot (\mathbf{r}_{ij} + \mathbf{n})][\boldsymbol{\mu}_j \cdot (\mathbf{r}_{ij} + \mathbf{n})]}{\mid \mathbf{r}_{ij} + \mathbf{n} \mid^5} \right\}. \tag{48}$$

where $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$. The sum over $\mathbf{n}$ is over all simple cubic lattice points, $\mathbf{n} = (n_x L, n_y L, n_z L)$ with n_x, n_y, n_z integers. The slowly decaying long range part of the dipolar potential makes the straightforward summation of Eqn.(48) too time consuming. The Ewald trick splits the problem again into two rapidly convergent parts, one in real space and one in reciprocal space. The details of the method are discussed in Refs.[3,5,7,15], here we only give the final expressions

$$U = U^{(r)} + U^{(k)} + U^{(s)} + U^{(d)}, \tag{49}$$

where the real-space $U^{(r)}$, the k-space (reciprocal-space) $U^{(k)}$, the self $U^{(s)}$ and the dipole (surface) $U^{(d)}$ contributions are respectively given by:

$$U^{(r)} = \frac{1}{2} \sum_{\mathbf{n}\in\mathbb{Z}^3}{}' \sum_{i,j=1}^{N} \Big\{ (\boldsymbol{\mu}_i \cdot \boldsymbol{\mu}_j) B(\mid \mathbf{r}_{ij} + \mathbf{n} \mid) - [\boldsymbol{\mu}_i \cdot (\mathbf{r}_{ij} + \mathbf{n})][\boldsymbol{\mu}_j \cdot (\mathbf{r}_{ij} + \mathbf{n})] C(\mid \mathbf{r}_{ij} + \mathbf{n} \mid) \Big\}, \tag{50}$$

$$U^{(k)} = \frac{1}{2L^3} \sum_{\mathbf{k}\in\mathbb{Z}^3, \mathbf{k}\neq 0} \frac{4\pi}{k^2} e^{-(\pi k/\alpha L)^2} \sum_{i,j=1}^{N} (\boldsymbol{\mu}_i \cdot \mathbf{k})(\boldsymbol{\mu}_j \cdot \mathbf{k}) e^{2\pi i \mathbf{k}\cdot\mathbf{r}_{ij}/L}, \tag{51}$$

$$U^{(s)} = -\frac{2\alpha^3}{3\sqrt{\pi}} \sum_{i=1}^{N} \mu_i^2, \tag{52}$$

$$U^{(d)} = \frac{2\pi}{(2\epsilon' + 1)L^3} \sum_{i,j=1}^{N} \boldsymbol{\mu}_i \cdot \boldsymbol{\mu}_j. \tag{53}$$

The sums over i and j are for the particles in the central box and

$$B(r) = [\mathrm{erfc}(\alpha r) + (2\alpha r/\sqrt{\pi}) \exp(-\alpha^2 r^2)]/r^3, \tag{54}$$

$$C(r) = [3\mathrm{erfc}(\alpha r) + (2\alpha r/\sqrt{\pi})(3 + 2\alpha^2 r^2) \exp(-\alpha^2 r^2)]/r^5. \tag{55}$$

The inverse length α is the splitting parameter of the Ewald summation which should be chosen so as to optimize the performance. The form Eqn.(53) given for the surface correction assumes that the set of the periodic replications of the simulation box tends in a spherical way towards an infinite cluster and that the medium outside this sphere is an uniform dielectric with dielectric constant ϵ'[15,5]. The case of a surrounding vacuum corresponds to $\epsilon' = 1$ and the surface term vanishes for the metallic boundary conditions ($\epsilon' = \infty$).

In practical calculations, the infinite sums in Eqns.(50, 51) are truncated by only taking into account distances which are smaller than some real space cutoff r_c and wave vectors with a modulus smaller than some reciprocal space cutoff k_c. If $r_c \leq L/2$, the sum in real space (Eqn.(50)) reduces to the normal minimum image convention. The double sum over particles in $U^{(k)}$ can be replaced by a product of two single sums which is more suitable for numerical calculations.

The force $\mathbf{F}_i$ acting on particle i is obtained by differentiating the potential energy U with respect to $\mathbf{r}_i$, i.e.,

$$\mathbf{F}_i = -\frac{\partial U}{\partial \mathbf{r}_i} = \mathbf{F}_i^{(r)} + \mathbf{F}_i^{(k)}, \tag{56}$$

with the real-space and k-space contributions given by:

$$\mathbf{F}_i^{(r)} = \sum_{\mathbf{n}\in\mathbb{Z}^3}{}' \sum_{j=1}^{N} \Bigg\{ \{(\boldsymbol{\mu}_i \cdot \boldsymbol{\mu}_j)(\mathbf{r}_{ij}+\mathbf{n}) + \boldsymbol{\mu}_i[\boldsymbol{\mu}_j \cdot (\mathbf{r}_{ij}+\mathbf{n})] + [\boldsymbol{\mu}_i \cdot (\mathbf{r}_{ij}+\mathbf{n})]\boldsymbol{\mu}_j\} C(\mid \mathbf{r}_{ij}+\mathbf{n} \mid)$$
$$- [\boldsymbol{\mu}_i \cdot (\mathbf{r}_{ij}+\mathbf{n})][\boldsymbol{\mu}_j \cdot (\mathbf{r}_{ij}+\mathbf{n})] D(\mid \mathbf{r}_{ij}+\mathbf{n} \mid)(\mathbf{r}_{ij}+\mathbf{n}) \Bigg\}, \tag{57}$$

$$\mathbf{F}_i^{(k)} = \frac{2\pi}{L^4} \sum_{j=1}^{N} \sum_{\mathbf{k}\in\mathbb{Z}^3, \mathbf{k}\neq 0} \frac{4\pi \mathbf{k}}{k^2} (\boldsymbol{\mu}_i \cdot \mathbf{k})(\boldsymbol{\mu}_j \cdot \mathbf{k}) \exp[-(\pi k/\alpha L)^2] \sin(2\pi \mathbf{k} \cdot \mathbf{r}_{ij}/L), \tag{58}$$

where

$$D(r) = [15\mathrm{erfc}(\alpha r) + (2\alpha r/\sqrt{\pi})(15 + 10\alpha^2 r^2 + 4\alpha^4 r^4) \exp(-\alpha^2 r^2)]/r^7. \tag{59}$$

Since the self and surface energy terms [Eqns.(52, 53)] are independent of the particle positions, they have no contributions to the force, unlike the Ewald summation for the Coulomb systems where the surface term contributes.

The torque $\boldsymbol{\tau}_i$ acting on particle i is related to the electrostatic field $\mathbf{E}_i$ at the location of this particle,

$$\boldsymbol{\tau}_i = \boldsymbol{\mu}_i \times \mathbf{E}_i = \boldsymbol{\tau}_i^{(r)} + \boldsymbol{\tau}_i^{(k)} + \boldsymbol{\tau}_i^{(d)}, \tag{60}$$

with

$$\mathbf{E}_i = -\frac{\partial U}{\partial \boldsymbol{\mu}_i}, \tag{61}$$

and thus

$$\boldsymbol{\tau}_i^{(r)} = -\sum_{\mathbf{n}\in\mathbb{Z}^3}{}' \sum_{j=1}^{N} \Bigg\{ (\boldsymbol{\mu}_i \times \boldsymbol{\mu}_j) B(\mid \mathbf{r}_{ij}+\mathbf{n} \mid) -$$
$$[\boldsymbol{\mu}_i \times (\mathbf{r}_{ij}+\mathbf{n})][\boldsymbol{\mu}_j \cdot (\mathbf{r}_{ij}+\mathbf{n})] C(\mid \mathbf{r}_{ij}+\mathbf{n} \mid) \Bigg\}, \tag{62}$$

$$\boldsymbol{\tau}_i^{(k)} = -\frac{1}{L^3} \sum_{j=1}^{N} \sum_{\mathbf{k}\in\mathbb{Z}^3, \mathbf{k}\neq 0} \frac{4\pi}{k^2} (\boldsymbol{\mu}_i \times \mathbf{k})(\boldsymbol{\mu}_j \cdot \mathbf{k}) e^{-(\pi k/\alpha L)^2} e^{2\pi i \mathbf{k} \cdot \mathbf{r}_{ij}/L}, \tag{63}$$

$$\boldsymbol{\tau}_i^{(d)} = -\frac{4\pi}{(2\epsilon'+1)L^3} \sum_{j=1}^{N} \boldsymbol{\mu}_i \times \boldsymbol{\mu}_j. \tag{64}$$

5.1 Error Formulas

We now give estimates for the rms error caused by cutting off the Ewald summation in real-space and k-space for the forces, the total energy and the torques. There are no errors involved in the self and surface contributions (Eqns(52, 53, 64)), because no cutoff operations are applied to them.

As can be shown similarly to the way as it was presented in Sec.2.4 the rms error for the force can be cast in the following form

$$\Delta F^{(r)} \approx \chi^{(r)} \frac{\mathcal{M}^2}{\sqrt{N}}, \tag{65}$$

where the quantity $\mathcal{M}$ is defined as

$$\mathcal{M}^2 := \sum_{j=1}^{N} \mu_j^2. \tag{66}$$

After some lengthy calculation one obtains

$$\chi^{(r)2} \approx L^{-3} r_c^{-9} \alpha^{-4} (\frac{13}{6} C_c^2 + \frac{2}{15} D_c^2 - \frac{13}{15} C_c D_c) \exp(-2\alpha^2 r_c^2), \tag{67}$$

where the terms C_c and D_c are given by

$$C_c = 4\alpha^4 r_c^4 + 6\alpha^2 r_c^2 + 3, \tag{68}$$

$$D_c = 8\alpha^6 r_c^6 + 20\alpha^4 r_c^4 + 30\alpha^2 r_c^2 + 15. \tag{69}$$

The resulting rms expectation of the real-space cutoff error in the forces is thus

$$\Delta F^{(r)} \approx \mathcal{M}^2 (L^3 \alpha^4 r_c^9 N)^{-1/2} (\frac{13}{6} C_c^2 + \frac{2}{15} D_c^2 - \frac{13}{15} C_c D_c)^{1/2} \exp(-\alpha^2 r_c^2). \tag{70}$$

The derivation of the expected real-space cutoff errors in the total potential energy and torques follows the same way. For calculating the fluctuation of the error in total energy, it is noted that the interaction energy between two dipoles is evenly shared between them. That means the sum of $\langle (\Delta U^{(r)})^2 \rangle$ over all particles contains each pair contribution twice and thus the fluctuation of the real-space cutoff error is one half of the sum[20]. Then the rms value of the real-space cutoff error of the total potential energy is estimated as

$$\Delta U^{(r)} \approx \mathcal{M}^2 (L^3 \alpha^4 r_c^7 N)^{-1/2} [\frac{1}{4} B_c^2 + \frac{1}{15} C_c^2 - \frac{1}{6} B_c C_c]^{1/2} \exp(-\alpha^2 r_c^2) \tag{71}$$

with

$$B_c = 2\alpha^2 r_c^2 + 1. \tag{72}$$

The rms error on the torques is estimated similarly to the force as

$$\Delta \tau^{(r)} \approx \mathcal{M}^2 (L^3 \alpha^4 r_c^7 N)^{-1/2} [\frac{1}{2} B_c^2 + \frac{1}{5} C_c^2]^{1/2} \exp(-\alpha^2 r_c^2). \tag{73}$$

Eqn.(70, 71, 73) all contain the exponential $\exp(-\alpha^2 r_c^2)$. For sufficiently low errors, αr_c has to be larger than one, for example $\alpha r_c \approx \pi$ for an error of $\exp(-\pi^2) \approx 5 \times 10^{-5}$. If

only the highest powers of αr_c are retained, the estimates of the real-space cutoff errors in the total energy, forces and torques can be reduced to

$$\Delta U^{(r)} \approx 4\mathcal{M}^2\alpha^2(r_c/15NL^3)^{1/2}\exp(-\alpha^2 r_c^2), \tag{74}$$

$$\Delta F^{(r)} \approx 8\mathcal{M}^2\alpha^4(2r_c^3/15NL^3)^{1/2}\exp(-\alpha^2 r_c^2), \tag{75}$$

$$\Delta \tau^{(r)} \approx 4\mathcal{M}^2\alpha^2(r_c/5NL^3)^{1/2}\exp(-\alpha^2 r_c^2), \tag{76}$$

where Eqn.(74) is a factor of $\sqrt{6/5}$ slightly larger than that given in Eqn.(35) of Ref.[20]. The advantage of these simplified formulas is that they reflect the dependence of the rms errors on α and r_c more directly and thus could be used more easily in determining the optimal values of these parameters.

In deriving the estimates of the reciprocal-space (k-space) cutoff errors, we further assume that the radial distribution function of the particles is approximately unity at all distances. Following Eqn.(58), the k-space cutoff error in the force acting on particle i is given by

$$\Delta\mathbf{F}_i^{(k)} = \sum_{j=1}^{N}\sum_{\mathbf{k},k>k_c}\frac{8\pi^2\mathbf{k}}{L^4k^2}(\boldsymbol{\mu}_i\cdot\mathbf{k})(\boldsymbol{\mu}_j\cdot\mathbf{k})\exp[-(\pi k/\alpha L)^2]\sin(2\pi\mathbf{k}\cdot\mathbf{r}_{ij}/L). \tag{77}$$

Note that the diagonal term ($j = i$) in the sum does not depend on the positions of the particles. It will provide a systematic contribution to the cutoff error in k-space[20]. In Eqn.(77) this contribution equals to zero, thus there is no systematic part of the error in the forces. The same thing happens to the error in the torques. But for the total energy the diagonal terms are positive and the systematic contribution plays a dominant role in the cutoff error.

The off-diagonal terms in Eqn.(77) do depend on the positions of the particles and have alternating signs. Similarly as before, the off-diagonal contribution to the cutoff error in $\Delta\mathbf{F}_i^{(k)}$ is given by

$$\Delta\mathbf{F}_{i,off}^{(k)} = \mid\boldsymbol{\mu}_i\mid\sum_{j\neq i}\mid\boldsymbol{\mu}_j\mid\boldsymbol{\chi}_{ij}^{(k)} \tag{78}$$

with

$$\boldsymbol{\chi}_{ij}^{(k)} = \sum_{\mathbf{k},k>k_c}\frac{8\pi^2\mathbf{k}}{L^4}\cos\vartheta(\hat{\boldsymbol{\mu}}_i,\hat{\mathbf{k}})\cos\vartheta(\hat{\boldsymbol{\mu}}_j,\hat{\mathbf{k}})\exp[-(\pi k/\alpha L)^2]i\exp(2\pi i\mathbf{k}\cdot\mathbf{r}/L), \tag{79}$$

where $\mathbf{r}$ stands for $\mathbf{r}_{ij}$, $\sin(2\pi\mathbf{k}\cdot\mathbf{r}/L)$ is re-written as $i\exp(2\pi i\mathbf{k}\cdot\mathbf{r}/L)$ according to the symmetrical character of the summation over $\mathbf{k}$, and $\hat{\mathbf{k}}$ is the unit vector along $\mathbf{k}$. Since the particles are assumed to be randomly distributed over the simulation box, the fluctuation of $\Delta\mathbf{F}_{i,off}^{(k)}$ can also be written as

$$\left\langle(\Delta\mathbf{F}_{i,off}^{(k)})^2\right\rangle = \mu_i^2\sum_{j\neq i}\sum_{m\neq i}\mid\boldsymbol{\mu}_j\mid\mid\boldsymbol{\mu}_m\mid\langle\boldsymbol{\chi}_{ij}^{(k)}\cdot\boldsymbol{\chi}_{im}^{(k)*}\rangle \approx \mu_i^2\mathcal{M}^2\chi^{(k)2}, \tag{80}$$

where again $\chi^{(k)2}$ is independent of i and j.

Choosing the z-axis of the spherical coordinates (k,θ,ϕ) along the $\hat{\boldsymbol{\mu}}$ orientation, we find

$$\chi^{(k)2} \approx 128\pi^3L^{-6}\alpha^2k_c^3\exp[-2(\pi k_c/\alpha L)^2]/15. \tag{81}$$

The rms expectation of the k-space cutoff error in the forces is thus

$$\Delta F^{(k)} \approx 8\pi\mathcal{M}^2 L^{-3}\alpha(2\pi k_c^3/15N)^{1/2}\exp[-(\pi k_c/\alpha L)^2]. \tag{82}$$

Here the notation $\Delta F_{off}^{(k)}$ is replaced directly with $\Delta F^{(k)}$ due to the fact of no diagonal contribution.

The derivation of the off-diagonal parts of the cutoff errors in the total energy and torques proceeds in the same way. That the sum over $\langle(\Delta U_{i,off}^{(k)})^2\rangle$ contains each pair contribution twice has also been considered in the error estimate of the total energy. The results are given by

$$\Delta U_{off}^{(k)} \approx 4\mathcal{M}^2 L^{-2}\alpha(\pi k_c/15N)^{1/2}\exp[-(\pi k_c/\alpha L)^2], \tag{83}$$

$$\Delta \tau^{(k)} \approx 4\mathcal{M}^2 L^{-2}\alpha(\pi k_c/5N)^{1/2}\exp[-(\pi k_c/\alpha L)^2]. \tag{84}$$

$\Delta\tau^{(k)}$ is also used directly instead of $\Delta\tau_{off}^{(k)}$.

The diagonal (systematic) part of the cutoff error in the total energy can be written as

$$\begin{aligned}\Delta U_{diag}^{(k)} &= \frac{1}{2\sqrt{N}}\sum_{i=1}^{N}\sum_{\mathbf{k},k>k_c}\frac{4\pi}{L^3}\mu_i^2\cos^2\vartheta(\hat{\boldsymbol{\mu}}_i,\hat{\mathbf{k}})\exp[-(\pi k/\alpha L)^2]\\ &\approx \frac{2\pi}{L^3\sqrt{N}}\mathcal{M}^2\int_{k_c}^{\infty}\exp[-(\pi k/\alpha L)^2]k^2dk\int_0^{\pi}\sin\theta d\theta\int_0^{2\pi}\cos^2\vartheta(\hat{\boldsymbol{\mu}},\hat{\mathbf{k}})d\phi\\ &\approx \frac{4}{3}\mathcal{M}^2 L^{-1}\alpha^2 k_c N^{-1/2}\exp[-(\pi k_c/\alpha L)^2],\end{aligned} \tag{85}$$

where the sum is again approximated by an integral and an integral expansion formula[20] is used to get the final estimate. The total k-space cutoff error in the total energy is thus

$$\Delta U^{(k)} = \Delta U_{diag}^{(k)} + \Delta U_{off}^{(k)}. \tag{86}$$

Comparing Eqn.(85) with (83), it can be seen that the systematic part of the error is a factor of $\sim L\alpha k_c^{1/2}(\gg 1)$ larger than the statistical part. Hence the systematic contribution is dominant in the k-space cutoff error of the total energy.

Assuming that the real-space and reciprocal-space contributions to the error are independent, the total cutoff error in Ewald summation can be written as

$$\Delta\Theta = \sqrt{\Delta\Theta^{(r)^2} + \Delta\Theta^{(k)2}}, \tag{87}$$

where Θ stands for U, F and τ.

5.2 Optimization of Parameters

Now we show how to use the analytical formulas derived in Sec. 5.1 to determine the optimal values of α, r_c and k_c by which the required accuracy could be satisfied and the computation time is minimized. The detailed discussions on this subject can also be found in Refs.[7,31].

The overall computation time for computing the forces with the Ewald method is approximately given by[31]

$$\mathcal{T} = a_r N^2(r_c/L)^3 + a_k N k_c^3, \tag{88}$$

where the primitive overheads a_r and a_k highly depend on the implementation of the code and need to be determined by numerical experiments. As an example, we have carried out timing experiments on a DEC personal workstation (CPU 433MHz) using a standard Fortran 77 compiler. In the implementation the complementary error function and its derivative are calculated with table lookup and the reciprocal-space summation is optimized as in Refs.[7,15]. The linked-cell method is used to deal with the short-range forces (when doing simulations). The primitive overheads are then found roughly to be $a_r = 2.5\mu s$ and $a_k = 0.7\mu s$.

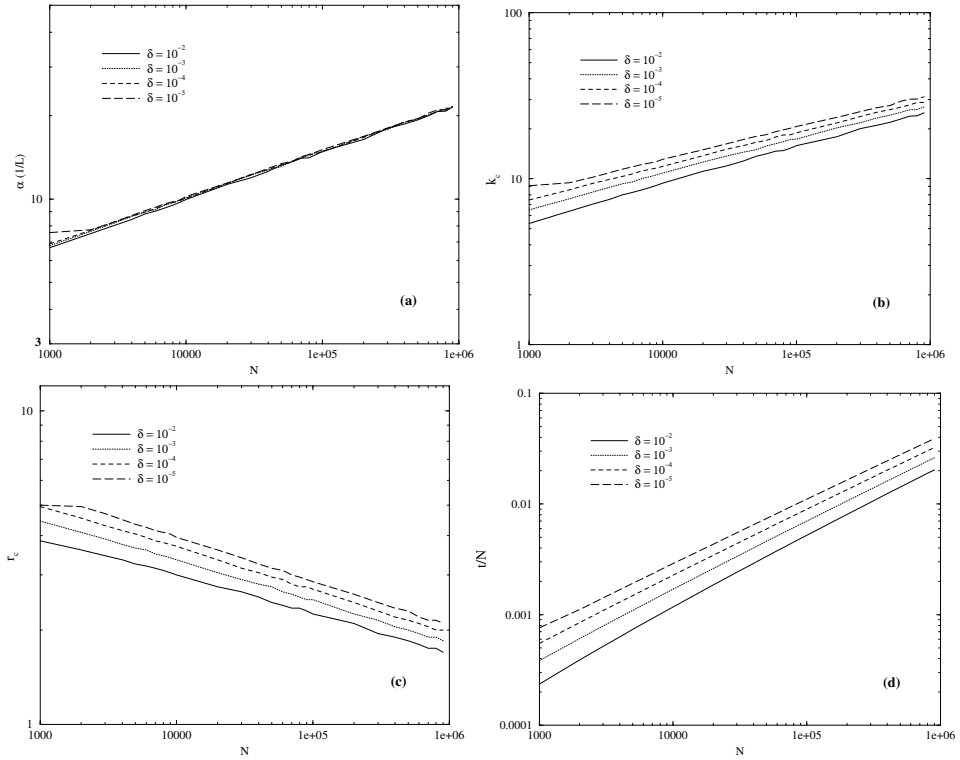


Figure 4. Optimal values of the parameters α (a), k_c (b) and r_c (c) as well as the corresponding minimized computation time $\mathcal{T}/N$ (d) as a function of the number of particles.

For a required accuracy δ, the parameters α, r_c and k_c should be chosen to minimize $\mathcal{T}$ with respect to the two constraints of the error bounds [Eqns.(70) and (82)], which are restated as

$$\frac{\delta}{\sqrt{2}} = \mathcal{M}^2(L^3\alpha^4 r_c^9 N)^{-1/2}(\frac{13}{6}C_c^2 + \frac{2}{15}D_c^2 - \frac{13}{15}C_c D_c)^{1/2}\exp(-\alpha^2 r_c^2), \quad (89)$$

$$\frac{\delta}{\sqrt{2}} = 8\pi\mathcal{M}^2 L^{-3}\alpha(2\pi k_c^3/15N)^{1/2}\exp[-(\pi k_c/\alpha L)^2]. \quad (90)$$

In case of $\delta \leq 5 \times 10^{-5}$, Eqn.(75) could be used instead of (70) so as to show the de-

pendence of the accuracy on the parameters more clearly. Eqn.(89) and (90) provide the qualitative function relations of r_c and k_c with α as: $r_c(\alpha) \approx -A\sqrt{\ln\delta}/\alpha$ and $k_c(\alpha) \approx -B\sqrt{\ln\delta}\alpha$. Inserting them into Eqn.(88) and differentiating it with respect to α yields $\alpha \propto N^{1/6}$ and thus $r_c \propto N^{-1/6}$ and $k_c \propto N^{1/6}$. The minimized computation time is then proportional to $N^{3/2}$ with the proportionality constant depending on the accuracy. The same results can be found for the Coulomb Ewald method in Refs.[7,20,31]. This can be easily understood by comparing Eqns.(70) and (82) in Sec. 5.1 of this paper with Eqns.(18) and (32) in Ref.[20], where one finds the same exponential dependences of the cutoff errors on α, r_c and k_c for the dipolar and Coulomb Ewald summations.

The numerical investigation of the functional dependence of the optimal parameters on N and δ are performed by using the primitive overheads obtained above. For each given N and δ, we at first choose different values for r_c within the inequality $r_c \leq L/2$. For each r_c the parameters α and k_c are calculated by solving Eqns.(89) and (90). These values are then introduced into Eqn.(88) to figure out the optimal value of r_c which gives the minimum computation time. In calculations the size of the simulation cell is fixed to a dimensionless length of $L = 10$. The range of accuracy requirement and number of particles are chosen to be $\delta = 10^{-2}$ to 10^{-5} measuring in $\mathcal{P}^2/\mathcal{L}^4$ and $N = 10^3$ to $N = 10^6$ which should cover most of the applications. The particles are supposed to have an uniform dipole moment of $\mathcal{P}$. The results for the optimal values of the parameters and the corresponding computation time per particle are shown in Fig. 4 (a-d), respectively. It can be clearly seen that the functional dependence of the parameters and the overall computation time on N are just as discussed above. Fig.4(c) shows that when a high accuracy is required for a system with a small number of particles, the predicted real-space cutoff is larger than half of the box length and $r_c = L/2$ must be used. The optimal α values hardly depend on the accuracies. These results are very similar to that obtained for the Coulomb Ewald summation in Ref.[31], except for $r_c \propto N^{-1/6}$ here and $r_c \propto N^{1/6}$ there. This is because they considered a system of constant density, while we choose the volume of the simulation cell to be constant.

6 The ESPResSo Project

As shown before, the algorithms for various fully or partially periodic geometries can be quite complex, and thus the production of single data points can take weeks or even more for complex biomolecular problems like protein folding, see for example other contributions in this volume. Not only data production takes time, also newcomers to the field experience often a painfully long learning period to get familiar with the intricacies of the methods and the code. This in turn implies, that modifications of the algorithms are difficult and time consuming. Considerable amounts of valuable research time is spent on coding issues with the final result of a highly specialized research tool. Often this results in human resources being kept away either from performing algorithmic improvements or scientific investigations. Since our group has been developing fast algorithms for quite some time now, mostly in specialized programs, we wanted to bundle them into one package. Moreover, other algorithms for soft matter research, coming from either colleagues at the theory group at the MPI-P, or interested research groups around the world, should possibly be included in the future. Our requirements therefore were, that the program should be easy to use, but scientifically sound; it should grant experts access to state-of-the-art techniques, but enable beginners to become experts as well, therefore it should be well-documented

providing exhaustive informations. And finally, new algorithmic additions should be fairly easy to be included, and certainly the program should not be a "black box".

Looking at other available simulation packages, e.g BALL[66], GISMOS[67], GROMOS[68], LAMMPS[69], NAMD[70], polyMD[71], and OCTA[72], we did not find a single package which met all our needs.

This led us to design a newly structured program for research on soft matter, which we called ESPResSo : an **E**xtensible **S**imulation **P**ackage for **Res**earch on **So**ft Matter Systems[73]. The program is intended to enable us to study soft matter model systems via MD and MC algorithms, with particular emphasis on extensibility for new, highly complex force/energy algorithms. Since the problems under investigation are located along scientific frontiers, meaning they are complex and computationally time-consuming, the program is parallelizable, fast, accurate, and easily modifiable. Here, we present only a short overview of the ESPResSo-package. Updates and more documentation can be found on the web page http://www.espresso.mpg.de/. The distribution of the source code adheres to the open source standards and can obtained upon request. We hope that other researchers can test, improve or even enlarge our code into a valuable research tool for the soft matter community.

6.1 Design

The ESPResSo design was developed to specifically serve the demands of our computational research group. While most simulation programs focus only on single aspects of such a project, ESPResSo is suited to help researchers in the *whole* process between the specification of a scientific problem and the interpretation of the results. Since ESPResSo offers a variety of methods and combines the knowledge of tens of man-years of research expertise on soft matter it helps newcomers to get into simulation techniques and to choose the right method for a certain problem. On the other hand experts can easily implement their own special routines into the framework of ESPResSo, enabling them to explore paths outside the scope of their own programs. Often new problems require new algorithmic solutions. ESPResSo helps the user implement new features due to its hierarchical structure, its modularity, its general data structures and its well-defined interfaces. A testsuite which is part of ESPResSo helps in checking if new features reproduce well-known physical properties of model systems. The hierarchical structure is well-suited for running simulations without required knowledge of the whole program package. The whole system setup is contained within a Tcl[74] script. A large number of sample scripts for various simulation problems help in developing new applications. Inside a simulation script, one can handle the entire simulation process from the specification of a system, the actual simulation, its analysis and the graphical output of the results. We want to emphasize our goal to design a practical research tool which is easy to learn, use and extend. This is especially supported by ESPResSo being a team project, since this ensures that every part of the code has to pass through a discussion process provoking a simple, effective and understandable implementation.

A difficult task in the design process arises from conflicts between different requirements regarding the simultaneous optimization of several aspects. In order to ensure that new researchers do not need too much time to learn how ESPResSo works, the code has to be kept simple, which is sometimes in contradiction to code optimization for computational speed. Aiming at being able to handle a wide variety of topics instead of solving only

specific problems leads to the same challenge of countering the code's tendency towards more complexity, hence less understandability.

Hierarchical program structure: ESPResSo is built up using three hierarchical program levels. The steering of the program is done on a script language level. All tasks are implemented as extensions to the script language dealing with input and output of data, setting of particle properties, interactions and parameters, and performing the integration and analysis of a given system. The basic simulation level is implemented in C. It contains the integrator as well as the calculation of fundamental observables like forces, torques, energies, pressure and temperature. These first two levels build up the part that is common to all investigated problems. Consequently, it is the part which should be known by a researcher using ESPResSo. Therefore, special emphasis was placed on simplicity and readability. The third level, also implemented in C, ensures the speed, efficiency and great generality of ESPResSo. This includes algorithms to accelerate the force and energy calculations used by the integrator as well as special algorithms to treat long-range interactions. Parallelization of all time-critical parts of the program enable efficient large scale simulations (see the benchmarks in Section 6.2). On this level, one also finds all implemented potentials for the particle interactions, and interfaces to other programs like VMD[75] for on-the-fly visualization of simulations.

Modularity: The hierarchical structure is accomplished by splitting the levels into different modules, which subdivides the otherwise large program package into manageable pieces. This is particularly important for the aspect of extensibility because it ensures that an extension does not affect the entire package but rather one or few modules. It also allows the user to concentrate on understanding those modules and their scientific background that are actually used for the particular problem under investigation.

Generality: To serve as a general research tool, ESPResSo needs to be able to handle a wide variety of problems. This includes different topologies, short- and long-range interactions, external fields, constraints, different boundary conditions, and various methods like MD or MC. In order to treat large scale simulations, it is also necessary to have an efficient parallelized code which runs on multiple CPU architectures, The abilities of the parallelization is demonstrated in 6.2.

The connectivity between the particles, which in other programs is often stored as a global topology, is incorporated locally at every particle. In this way the user is free specifying the needed topologies on the script level, and the connectivity information can easily be parallelized. Since this allows for any kind of topology, additional concepts such as molecules, polymer chains or proteins are unnecessary. There are however a number of predefined sample scripts and auxiliary routines provided for the user's convenience which set up polymer chains, simple model networks, or more complex structures such as fullerenes, representing tutorial-like shortcuts that facilitate writing new task scripts. Since both single molecule experiments as well as investigations of confined systems have gained increasing importance over the last years, ESPResSo also contains features to handle corresponding simulations. The program is able to deal with periodic boundary conditions in any combination of up to three spatial directions using any of the previously described methods. When simulating bulk systems, e.g. a small representative portion of a solution, normally periodic boundary conditions are applied to avoid boundary effects. For simulations of thin films or surface effects, periodic boundary conditions in all three spatial dimensions do no make sense. The proper boundary conditions are periodic only in

two out of the three spatial dimensions, while the remaining coordinate has a non periodic boundary condition. If rods are the object of interest, only one coordinate is left to have periodic boundary conditions. As we have elucidated in the previous sections, the complexity of an electrostatic simulation dramatically changes with different boundary conditions.

In the case of a thin film, the particles have to be confined to a fixed layer. For this ESPResSo supports constraints like walls, cylinders or spheres. It is also possible to simulate particles subject to external forces or fields. In order to cover a wide range of thermodynamic environments one can switch between simulating different thermodynamic ensembles like the NVE-, NVT- or NPT-ensemble.

Documentation: ESPResSo is not intended to be a black-box-like package. Users are encouraged to try to understand its algorithms and routines, developers are strongly advised to do so before extending it. In order to preserve the knowledge about algorithms and their physical/chemical background, it is important to provide and maintain a well structured documentation. This is mainly done inside the code itself and then extracted and processed by doc++[76] into a user-friendly html-manual mainly addressing specific code-/function-/procedure-related issues. It is supplemented by a stand-alone documentation on general topics such as the usage of the script commands, the general organization of the data structures, communication schemes, and analysis options. The code development itself is done in a CVS-environment (concurrent version system[77]). This helps to keep track of all changes, and provides information on what, when and by whom something has changed in the program.

Programming environment: We decided to use C as the only programming language in order to keep the code as simple to read as possible. Compared to C++ we think that this is still the language of choice in a research environment since it is easier to learn for people having a natural science rather than a computer science background. At the same time it provides all necessary features to create a modular and concise program package.

We use Tcl[74] as the script language since it contains a simple and effective interface to build C programs as extensions to the script language itself. Syntax and programming style are similar to C which makes it easy to learn. Another advantage is that there exist a large variety of extensions for Tcl. For example, with the Tk-extension it is straight forward to build a graphical user interface. This has already been done for presentation purposes. Tcl also gives us the possibility to easily create interfaces to other programs. Examples are gnuplot or xmgr for graphical processing of analysis results.

Another important choice was the type of communication for the parallelization. We decided on using MPI, as it is available for virtually all architectures; unlike e.g. OpenMP which requires shared memory, MPI also works on distributed memory computers such as Linux clusters. ESPResSo relies on the fact that MPI–implementations are normally well optimized for the underlying architecture.

Ease of use: For the ESPResSo-package to live up to its full potential, a straightforward and simple access is mandatory. The layered hierarchical program structure allows the user to focus on any aspect of his scientific simulation. This can be for example modeling complex physical systems with particle insertion/deletion, pressure-dependent volume changes and/or varying constraints can be done by simply creating a corresponding script file which specifies the basic rules of such a setup. Or it can be tweaking computational routines to utmost performance can be achieved by simply adding/modifying/replacing one single module of ESPResSo, immediately granting other users access to that im-

provement.

6.2 Benchmarks

Even though the primary goals of the ESPResSo-package are accessibility, modularity, flexibility, and extensibility, its secondary -and equally important- objective accounts for the scientific realities of tight schedules and time constraints: Being as optimized for speed as possible without sacrificing its primary benefits. Considering the timings we measured, the state-of-the-art algorithms adapted meet both demands extremely well. The direct comparison of benchmark timings of some test scenarios to those of the corresponding highly specialized codes show that despite the unique nature of ESPResSo representing a very general multi-purpose tool, it still performs similarly (e.g. compared to LAMMPS, but around 1.5 times slower than polyMD) with a firm robustness among different architectures (e.g. AMD, Intel, IBM, Alpha), wrapper (e.g. mpicci, mpich, mpicc), and operating systems (e.g. AIX, OSF1, Linux). To our knowledge there is no other available program package similar in scope and design of ESPResSo. Any potential difference in performance to highly specialized codes is negligible compared to either the time to be spend for producing even faster code (particularly when recalling ESPResSo's goal to remain understandable, prohibiting most low-level trickery), or to the advancements in hardware technology, or simply overcompensated for by ESPResSo's inherent design advantages since e.g. the modularity allows for an easy incorporation of any algorithmical improvements which might arise in the future.

Besides absolute speed, another fundamental feature of our scientific simulation system is its intrinsic parallelizability, which also distinguishes it from other projects: In ESPResSo the choices of data structures and algorithm implementations were optimized in this respect, so that the program is now able to use any reasonable number of processors on any computer system supporting one of the available MPI–environments. To demonstrate the parallel performance, Table 1 presents benchmarking results for three standard test scenarios. Note that the charged systems used the P^3M routines, which is why for

# of Processors N	1	2	4	8	16	32
LJ-system (fixed)	1.0000	0.9888	0.9743	0.9587	0.9171	0.8437
LJ-system (scaled)	1.0000	0.9873	0.9726	0.9166	0.8913	0.8211
dense ES (fixed)	1.0000	0.9738	0.9475	0.8817	0.8115	0.6946
dense ES (scaled)	1.0000	0.9637	0.9955	0.7609	0.7446	0.5033
dilute ES (fixed)	1.0000	0.8722	0.8307	0.7343	0.5992	0.4594
dilute ES (scaled)	1.0000	0.8574	0.6823	0.6597	0.6080	0.3549

Table 1. Efficiency of the ESPResSo-code on an IBM Regatta H Server (eServer 690 Model 681 with 32 Power4 Processors at 1.3 GHz each) for three different systems: A neutral LJ fluid composed of either $32000 \cdot N$ (scaled) or 32000 particles (fixed) at a density of 0.8442 in a NVE-ensemble; a dense electrolyte system with either $2000 \cdot N$ (scaled) or 8000 particles (fixed) at a density of 0.07, friction and temperature of 1.0, Bjerrum length of 20.0 in a NVT-ensemble; a dilute electrolyte system which differs only in density ($1 \cdot 10^{-4}$), Bjerrum length (2.0), and fixed size (16000 particles). After warm-up and equilibration period the execution time T_N for integrating 1000 time steps (`integrate 1000`) was measured and compared between using one and N nodes: For a fixed-size system the efficiency reads $\frac{T_1/N}{T_N}$, for scaled-size $\frac{T_1}{T_N}$.

$N = 4$ the scaled dense ES-system seems more efficient compared to the scaled case at $N = 2$ or to the fixed system because the performance critically depends on the choice of P^{3}M-parameters whose optimizability in turn depends on the number of particles and processors.

6.3 Further Developments

As of this writing the ESPResSo-package continues to undergo significant enlargement. We are currently implementing a standard dipolar Ewald sum[65], which will enhance the capabilities to simulate ferrofluids or dipolar fluids like simple water models, and add an enhanced leap-frog algorithm for the rotational degrees of freedom. Also work has started to include an anisotropic short range potential, namely the Gay-Berne potential, which will allow the study of liquid crystals.

For the dynamics of soft matter systems it is often necessary to include hydrodynamic interactions. Since in practice one cannot include all molecular details of the systems this can be achieved on a coarse grained level by coupling the solvent degrees of freedom to the simulated particles. This will be implemented via an advanced lattice Boltzmann algorithm from the group of B. Dünweg that has already proven its usefulness in polymer dynamics simulations[78]. An alternative way of coarse graining hydrodynamics, called dissipative particle dynamics (DPD)[79], is based on a momentum conserving thermostat. We will implement a version according to Soddemann et al.[80]. And finally, a non-equilibrium molecular dynamics algorithm[81], especially useful for driven systems, will also be added.

On top of the present strengths of ESPResSo concerning the efficient treatment of electrostatics, we plan to implement two very new ideas which promise to be a significant improvement in investigating media with varying local dielectric constants: While the first is a purely local algorithm by T. Maggs[82] which seems to be very well suited for MC simulations and can also be very useful for dense systems by using a constrained MD algorithm[c], the second is a finite difference multigrid scheme for electro- and magnetostatics[d] which appears to be better for parallel applications, promising to be fast for MD algorithms as well due to its recursiveness.

There are many more improvements planned for the next year, and hopefully the capabilities of ESPResSo will grow even further once other researchers take up our idea and contribute to ESPResSo by using, customizing and extending it. More information, and the latest version can be found at http://www.espresso.mpg.de/

7 One Application: Polyelectrolytes in Poor Solvent

In the last section we want to present one application of a soft matter system, namely a polyelectrolyte system, where charges play a dominant role. Polyelectrolytes (PEs) are polymers which have the ability to dissociate charges in polar solvents resulting in charged polymer chains (macroion) and mobile counterions. They represent a broad and interesting class of soft matter[83,2] that enjoy an increasing attention in the scientific community. In technical applications PEs are used as viscosity modifiers, as precipitating agents, and as

[c] B. Dünweg, private communication
[d] I. Tsukerman – private communication

superabsorbers. A thorough understanding of charged soft matter has become of great interest also in biochemistry and molecular biology. This is due to the fact that virtually all proteins, as well as other biopolymers such as DNA, actin, or microtubules are PEs.

Many PEs have a hydrocarbon based backbone for which water is a very poor solvent. Therefore, in aqueous solution, there is a competition between solvent quality, Coulombic interaction, and entropic degrees of freedom. The conformation of individual chains can under certain conditions assume pearl-necklace like structures[84–86]. In earlier simulations[87,88] we found that the polymer density can be used as a very simple parameter to separate different conformational regimes. Here we analyze in more detail the single chain behavior and the scaling of the peak in the inter-chain structure function.

7.1 Simulation Model

Our PE model and MD approach has been described in Refs.[87,88,29,89,90] and consists of a bead spring chain of Lennard-Jones (LJ) particles. Chain monomers interact via a LJ interaction up to a distance $R_c = 2.5\sigma$ and experience an attraction with $\epsilon = 1.75\, k_B T$. The Θ-point for this model is at $\epsilon = 0.34 k_B T$[87]. The counterions interact via a purely repulsive LJ interaction. For bonded monomers we add a FENE bond potential. Charged particles at separation r interact via the Coulomb energy $k_B T \ell_b q_i q_j / r$, with $q_i = 1, (-1)$ for the charged chain monomers (counterions) and the Bjerrum length $\ell_b = e^2/(4\pi\epsilon_S\epsilon_0 k_B T)$ (e: unit charge, ϵ_0 and ϵ_S: permittivity of the vacuum and of the solvent). We simulated various systems with several chains in the central simulation box at various monomer densities ρ_m and different values of ℓ_b. Each chain consists of $N_m = 48 \ldots 478$ monomers, with a charge fraction $f = 1 \ldots 1/3$. The pressure p was found to be always positive, with the pV diagram being convex at all densities, thus our simulations are stable, reach true thermal equilibrium, and reside in a one phase region.

7.2 Single Chain Properties

With the help of a specially developed cluster algorithm[29,90] that automatically recognizes the number of pearls in a conformation, we have analyzed all equilibrium conformations in our systems. We found large coexistence regimes between structures consisting of conformations with different pearl numbers. Even a single chain shows over the course of time many transitions between different pearl structures, hence the different pearl states are not frozen or metastable. Also the position and size of the pearls and strings is constantly changing[29,89], compare Fig.5. We furthermore found, that the lower the pearl number, the stronger the counterions are attracted to the pearls. This is easy to rationalize, since smaller number of pearls mean larger pearls due to mass conservation, and thus to a higher local charge density. The integrated ion distribution versus the chain distance displays an inflection point, which is a signal of counterion condensation[54]. In contrast to analytical theories[91], the pearl structures are stable, even though there are counterions localized near and/or inside the pearls.

When one starts in a necklace conformation and increases ℓ_b the counterions will get attracted more and more towards the chain. Scaling theories have predicted that with the onset of condensation the necklace state should collapse in a first order transition into the globular state[92,86,91,93]. However, the 'onset' of condensation is not a sharp border, rather

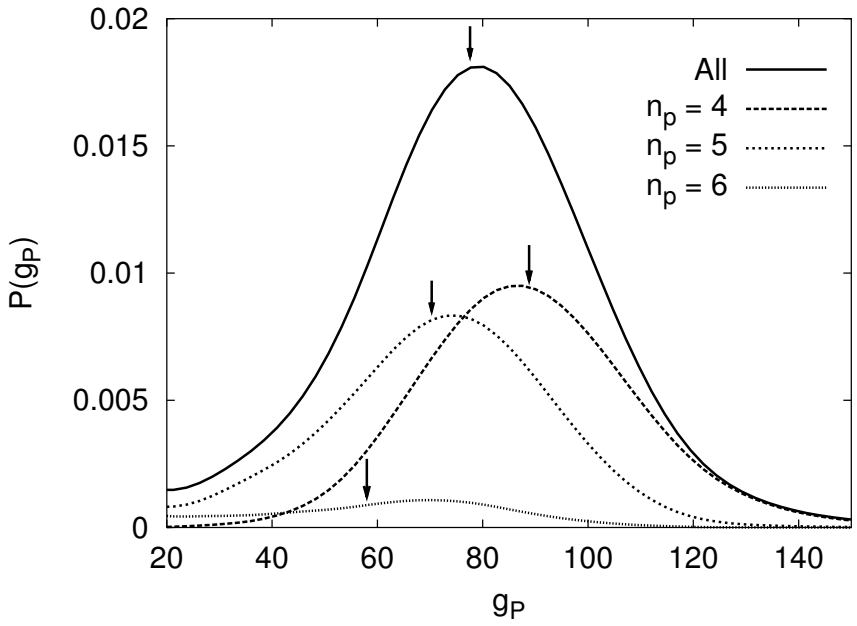


Figure 5. Probability distributions P for the pearl size g_P for a system with chain length $N_m = 382$, $\ell_b = 1.5\sigma$, and $f = 1/3$. Shown is the distribution for all chains as well as the distributions for the different structure types. The arrows mark the mean value of the corresponding probability distribution.

like within Poisson-Boltzmann theory one finds a smooth distribution of counterions which gets weighted closer to the macroion as the coupling is increased[54]. Accordingly, in the simulation we do not observe a collapse transition. The picture is qualitatively the same as in the good solvent case[94]. At $\ell_b = 0$ (no electrostatic interaction) the chain is in a collapsed conformation. By increasing ℓ_b the chain first extends up to a maximum, and then slowly shrinks back to a collapsed state. The non-monotonic behavior of the extension is qualitative the same as in the good solvent case[94] however the decrease is faster and more pronounced here[87,95]. There is also a subtle dependence on f. The scaling variable which determines R_E of the necklace is $f^2\ell_b$ at fixed N_m and ϵ[86]. In Fig.6 we show snapshots of chains with the same value of $f^2\ell_b$, but different f. The chain extension is drastically different, and depends on the local interactions mediated by the counterions. This effect is obviously not captured by the scaling Ansatz! Note also, that the conformations on the way to the collapsed globule are very much reminiscent of a cylindrical shape[92], since the strings become very short, and the pearls coalesce slowly on a sausage like string until they reach the globular state. The collapsed state is reached roughly at the same value of ℓ_b, which is reminiscent of the critical point of a Coulomb fluid.

Next we computed the spherically averaged form factor $S_1(q)$ of a single chain, shown in Fig.7, since this is an observable that is accessible in experiments, and for which also theoretical expressions have been developed[86,96,97].

In the range $1 < q\sigma < 2$ we denote a sharp decrease in S_1, which reflects the intra pearl scattering, because it shows the typical Porod scattering of $S_1(q) \simeq q^{-4}$. The kink

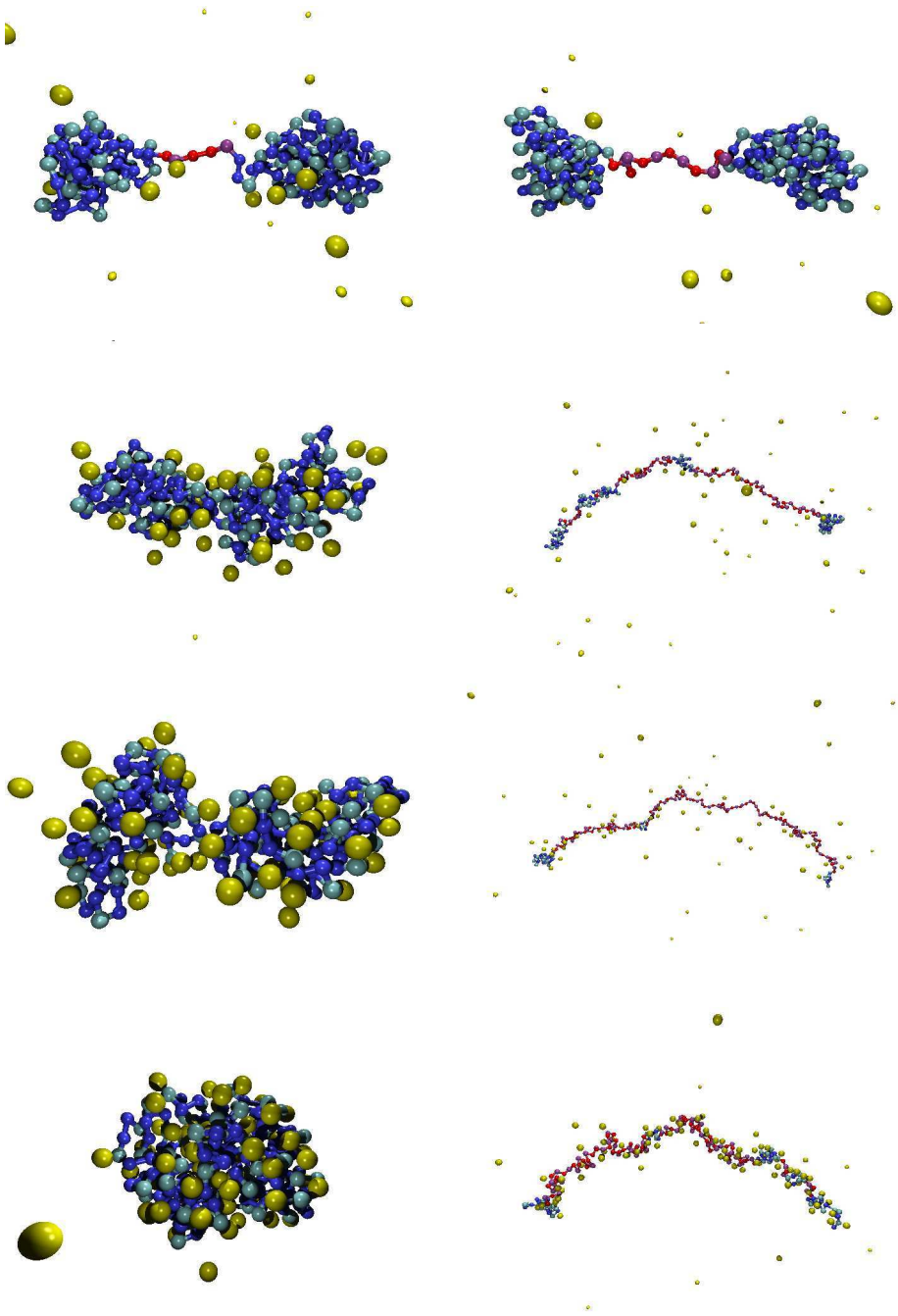


Figure 6. Snapshots for different values of the scaling variable $f^2\ell_b$. Left row with $f = 1/3$, right row with $f = 1/2$. From top to bottom $f^2\ell_b$ has the values: $0.08\sigma, 0.25\sigma, 0.5\sigma, 1.0\sigma$. System: 8 chains with $N_m = 199$ monomers at $\rho_m = 5.0 \times 10^{-5}\sigma^{-3}$.

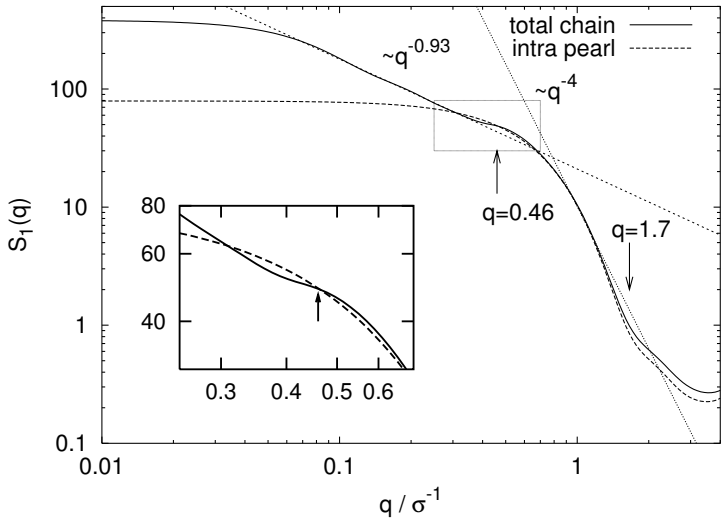


Figure 7. Spherically averaged form-factor $S_1(q)$. Shown are the single chain form-factor (solid line), together with the part of the form factor coming from the intra-pearl scattering (dashed line). The dotted and short dashed fits show the elongated chain part, and the Porod scattering part (globular conformation). System: 7 chains with 382 monomers, $f = 1/3$, $\ell_b = 1.5\sigma$, $\rho_m = 1.0 \times 10^{-5}\sigma^{-3}$.

at $q\sigma \approx 1.66$ appears at the position expected from the pearl size, but is broadly smeared out due to large size fluctuations. The shoulder which can be seen at $q\sigma \approx 0.5$ does not come from the intra-pearl scattering but is due to the scattering of neighboring pearls along the chain (inter-pearl contribution), which have a mean distance of $\langle r_{PP} \rangle = 13.3.\sigma$. It is also smeared out due to the large distribution of inter-pearl distances. We conclude that the signatures of the pearl-necklaces are weak already for monodisperse samples. A possible improvement could be achieved for chains of very large molecular weights and low pearl numbers, which could lead to stable and large signatures.

7.3 Solution Properties: Scaling of the Correlation Length ξ

The overall scattering function $S(q)$ of the solution contains additional experimental information. For good solvent PEs, experiments[98], theory[99], and simulations[94] find a pronounced first peak of $S_{IC} = S/S_1$ at $q^* = (2\pi)/\xi$, where ξ is the correlation length. The position varies as $q^* \propto \rho_m^{1/3}$ in the very dilute regime and crosses over to a $\rho_m^{1/2}$ regime at higher concentrations. In Fig. 8 we have plotted the density dependence of q^* in poor solvent for different chain lengths. Within the error bars we find that for poor solvent chains q^* scales proportional to $\rho_m^{0.35\pm0.04}$ for *all* concentrations and chain lengths. This is in accord with very recent experiments[100], but theoretically not well understood. The response of the PE conformation to density changes is much larger in the poor solvent case[87,29] than in the good solvent case[94], and the chain extension behaves non-monotonic as a function of density[87,29]. Furthermore, in the density regime between $\rho_m\sigma^3 = 10^{-2} \ldots 10^{-4}$ the chain extension and the pearl number varies most strongly, and almost all monomers

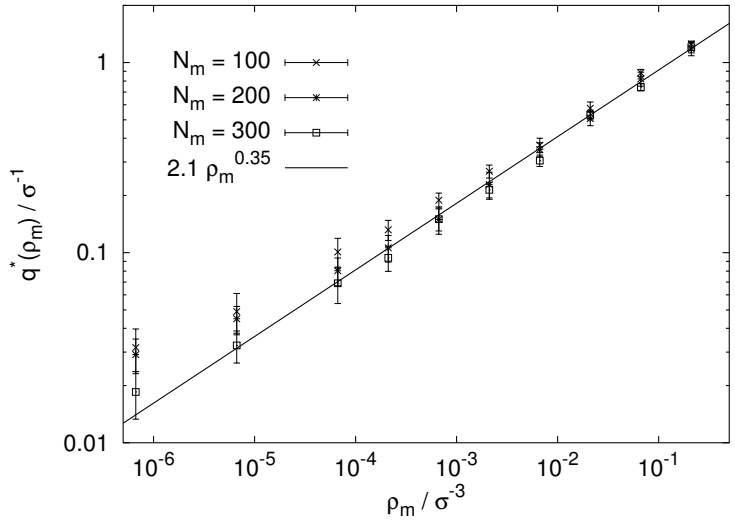


Figure 8. Density dependence of the peak q^* in the structure factor for three different chain length $N_m = 100, 200, 300$ with $f = 0.5$ and $\ell_b = 1.5\sigma$. The black line is a fit to the data with $N_m = 200$.

are located within the pearls. Upon approaching the dense regime, the string length tends to zero and we find a chain of touching pearls, indicating that the conventional necklace picture breaks down. Our result is compatible to scaling exponents found in scattering experiments[96,101,102].

Scaling theories[103,104] have predicted a $\rho_m^{1/2}$ regime to start at ρ_o^*, which is defined at the density where $R_E \approx \xi$, and to extend until $\xi \approx r_{pp}$ where a bead-controlled $\rho_m^{1/3}$ regime starts. We find $\rho_o^*\sigma^3 \simeq 5 \times 10^{-2}, 10^{-3}, 10^{-4}$ for $N_m = 100, 200, 300$. A pearl-pearl separation of the order of the correlation length length, $r_{pp} \approx \xi$, is reached between $\rho_m\sigma^3 = 10^{-2}$ and 10^{-1}, which is roughly independent of N. Especially for the longer chains ($N_m = 200, 300$) a clear signature of a different power law, i.e.$\rho_m^{1/2}$, should be visible. One possible reason for our different findings is that the strong inter-chain coupling and the influence of the counterions on the conformations are not sufficiently taken into account in the (mean field) scaling approach. It is not clear at this stage if the $\rho_m^{1/2}$ regime can be recovered for much longer chain length. In addition we observe that the chains form a transient physical network at $\rho_m\sigma^3 = 0.2$ for $N_m \geq 200$ which has neither been seen in previous simulations nor predicted by theoretical approaches but is in accord with experimental studies[96,101,102]. During the simulation time these networks reconstruct several times, e.g. chains are not trapped!

A more detailed account of the presented material can be found in the work of Limbach et al.[89,105,90,106,107].

8 Concluding Remarks

We have given a review of methods to compute long range interactions in fully or, for the case of the Coulomb sum, also partially periodic boundary conditions. All methods presented in more detail, have in common, that analytic error estimates exists, which allow easy tuning of the algorithms for speed and accuracy. For our research purposes, these methods are more or less the optimal choices at the moment. They are, or will be shortly, implemented in a newly designed program package, called ESPResSo which will be our main simulation program for the future. This program is open source, and informations regarding the latest version, or informations on how to participate in the further development of this package can be found on http://www.espresso.mpg.de/. Finally we have given a short summary of our recent findings about the conformational properties of polyelectrolytes in poor solvent.

Acknowledgments

This article would have been impossible without the contributions of A. Arnold, J. Dejoannis, M. Deserno, H.J. Limbach, B.A. Mann, and Z. Wang. Helpful discussions with K. Kremer and other members of our pep group are gratefully acknowledged. This work was supported by the "Zentrum für Multifunktionelle Werkstoffe und Miniaturisierte Funktionseinheiten", grant BMBF 03N 6500, and the DFG within the SFB 625, TR6, and grant HO-1108/11-1.

References

1. Richard A. L. Jones. *Soft Condensed Matter*. Oxford University Press, Great Clarendon Street, Oxford OX2 6DP, 2002.
2. Christian Holm, Patrick Kékicheff, and Rudi Podgornik, editors. *Electrostatic Effects in Soft Matter and Biophysics*, volume 46 of *NATO Science Series II - Mathematics, Physics and Chemistry*. Kluwer Academic Publishers, Dordrecht, Nl, December 2001.
3. P. P. Ewald. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Ann. Phys.*, 64:253–287, 1921.
4. D. M. Heyes. Electrostatic potentials and fields in infinite point charge lattices. *J. Chem. Phys.*, 74(3):1924–1929, February 1981.
5. Simon W. de Leeuw, John W. Perram, and E. R. Smith. Simulation of electrostatic systems in periodic boundary conditions. i. lattice sums and dielectric constants. *Proc. R. Soc. Lond. A*, 373:27–56, 1980.
6. Simon W. de Leeuw, John W. Perram, and E. R. Smith. Simulation of electrostatic systems in periodic boundary conditions. ii. equivalence of boundary conditions. *Proc. R. Soc. Lond. A*, 373:57–66, 1980.
7. J. Perram, H. G. Petersen, and Simon de Leeuw. An algorithm for the simulation of condensed matter which grows as the 3/2 power of the number of particles. *Mol. Phys.*, 65:875, 1988.

8. Markus Deserno and Christian Holm. How to mesh up Ewald sums. i. a theoretical and numerical comparison of various particle mesh routines. *J. Chem. Phys.*, 109:7678, 1998.
9. Markus Deserno and Christian Holm. How to mesh up Ewald sums. ii. an accurate error estimate for the p3m algorithm. *J. Chem. Phys.*, 109:7694, 1998.
10. M. Deserno. *Counterion condensation for rigid linear polyelectrolytes*. PhD thesis, Universität Mainz, 2000.
11. Axel Arnold and Christian Holm. Mmm2d: A fast and accurate summation method for electrostatic interactions in 2d slab geometries. *Comp. Phys. Comm.*, 148(3):327–348, 1 November 2002.
12. Axel Arnold, Jason de Joannis, and Christian Holm. Electrostatics in periodic slab geometries i. *J. Chem. Phys.*, 117:2496–2502, 2002.
13. Jason de Joannis, Axel Arnold, and Christian Holm. Electrostatics in periodic slab geometries ii. *J. Chem. Phys.*, 117:2503–2512, 2002.
14. Daan Frenkel and Berend Smit. *Understanding Molecular Simulation*. Academic Press, San Diego, second edition, 2002.
15. Mike P. Allen and Dominik J. Tildesley. *Computer Simulation of Liquids*. Oxford Science Publications. Clarendon Press, Oxford, 1 edition, 1987.
16. Jean-Michel Caillol. Comments on the numerical simulation of electrolytes in periodic boundary conditions. *J. Chem. Phys.*, 101(7):6080–6090, 1994.
17. Stefan Boresch and Othmar Steinhauser. Presumed versus real artifacts of the ewald summation technique: The importance of dielectric boundary conditions. *Ber. Bunsenges. Phys. Chem.*, 101(7):1019–29, 1997.
18. H. J. C. Berendsen. In Wilfred F. van Gunsteren, P. K. Weiner, and A. J. Wilkinson, editors, *Computer Simulation of Biomolecular Systems*, volume 2, pages 161–81, The Netherlands, 1993. ESCOM.
19. Philippe H. Hünenberger. Optimal charge-shaping functions for the particle-particle-particle-mesh (p3m) method for computing electrostatic interactions in molecular simulations. *J. Chem. Phys.*, 113(23):10464–10476, 2000.
20. Jiri Kolafa and John W. Perram. Cutoff errors in the ewald summation formulae for point charge systems. *Molecular Simulation*, 9(5):351–68, 1992.
21. Gerhard Hummer, Lawrence R. Pratt, and Angle E. García. Molecular theories and simulation of ions and polar molecules in water. *J. Phys. Chem. A*, 102(41):7885–97, 1998.
22. M. J. L. Sangster and M. Dixon. Interonic potentials in alkali-halides and their use in simulations of molten-salts. *Adv. Phys.*, 25:247, 1976.
23. D. J. Adams and G. S. Dubey. Taming the ewald sum. *J. Comp. Phys.*, 72:156, 1987.
24. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C*. Cambridge University Press, Cambridge, 2 edition, 1992.
25. R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles*. IOP, London, 1988.
26. T. Darden, D. York, and L. Pedersen. Particle mesh ewald: An $n \log(n)$ method for ewald sums in large systems. *J. Chem. Phys.*, 98:10089, 1993.
27. U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. Pedersen. A smooth particle mesh ewald method. *J. Chem. Phys.*, 103:8577, 1995.
28. E. L. Pollock and Jim Glosli. Comments on p^3m, fmm, and the ewald method for

large periodic coulombic systems. *Comp. Phys. Comm.*, 95:93–110, 1996.
29. Hans Jörg Limbach. *Struktur und Eigenschaften von Polyelektrolyten im schlechten Lösungsmittel*. PhD thesis, Johannes Gutenberg Universität, Mainz, Germany, 2001.
30. I. J. Schoenberg. *Cardinal Spline Interpolation*. Society for Industrial and Applied Mathematics, Philadelphia, 1973.
31. Henrik G. Petersen. Accuracy and efficiency of the particle mesh ewald method. *J. Chem. Phys.*, 103(9):3668–79, 1995.
32. J. Lekner. Summation of coulomb fields in computer simulated disordered systems. *Physica A*, 176:485–498, 1991.
33. R. Sperb. An alternative to ewald sums - part i: Identities for sums. *Molecular Simulation*, 20(3):179–200, 1998.
34. R. Sperb. An alternative to ewald sums, part 2: The coulomb potential in a periodic system. *Molecular Simulation*, 22(3):199–212, 1999.
35. R. Strebel and R. Sperb. An alternative to ewald sums, part 3: Implementation and results. *Molecular Simulation*, 27(1):61–74, 2001.
36. J. E. Barnes and P. Hut. A hierarchical o(n-log-n)force/calculationn algorithm. *Nature*, 324:446, 1986.
37. L.Greengard and V. Rhoklin. A fast algorithm for particle simulations. *J. Comp. Phys.*, 73:325, 1987.
38. K. Esselink. A comparison of algorithms for long-range interactions. *Comp. Phys. Comm.*, 87:375–395, 1995.
39. C. Sagui and T. Darden. Multigrid methods for classical molecular dynamics simulations of biomolecules. *J. Chem. Phys*, 2001.
40. I. Tsukermann. Flexible local approximation method for electro- and magnetostatics. In *IEEE Trans. Magn.*, to appear in 2004.
41. Albert H. Widmann and David B. Adolf. A comparison of ewald summation techniques for planar surfaces. *Comp. Phys. Comm.*, 107:167–186, 1997.
42. M. Kawata and U. Nagashima. Particle mesh ewald method for three-dimensional systems with two-dimensional periodicity. *Chem. Phys. Lett.*, 340:165–172, 2001.
43. Mazars M. Comment on "rapid calculation of the coulomb component of the stress tensor for three-dimensional systems with two-dimensional periodicity". *J. Chem. Phys*, 117(7):3524, Aug. 2002.
44. Mazars M. Lekner summations. *J. Chem. Phys*, 115(7):2955, Aug 2001.
45. Axel Arnold and Christian Holm. A novel method for calculating electrostatic interactions in 2d periodic slab geometries. *Chemical Physical Letters*, 354:324–330, 2002.
46. J. C. Shelley and G. N. Patey. Boundary condition effects in simulations of water confined between planar walls. *Mol. Phys.*, 88:385, 1996.
47. E. Spohr. Effect of electrostatic boundary conditions and system size on the interfacial properties of water and equeous solutions. *J. Chem. Phys.*, 107:6342, 1997.
48. In-Chul Yeh and Max L. Berkowitz. Ewald summation for systems with slab geometry. *J. Chem. Phys.*, 111(7):3155–3162, 1999.
49. P. Mináry, M. E. Tuckerman, K. A. Pihakari, and G. J. Martyna. A new reciprocal space based treatment of long range interactions on surfaces. *J. Chem. Phys.*, 116:5351, 2002.
50. E. R. Smith. Electrostatic energy in ionic crystals. *Proc. R. Soc. Lond. A*, 375:475–

505, 1981.
51. D. E. Parry. Electrostatic potential in surface region of an ionic-crystal. *Surf. Sci.*, 49:433, 1975.
52. D. E. Parry. Correction. *Surf. Sci.*, 54:195, 1976.
53. Axel Arnold. Berechnung der elektrostatischen wechselwirkung in 2d + h periodischen systemen. Diploma thesis, Johannes Gutenberg-Universität, may 2001.
54. Markus Deserno, Christian Holm, and Sylvio May. The fraction of condensed counterions around a charged rod: Comparison of Poisson-Boltzmann theory and computer simulations. *Macromolecules*, 33:199–206, 2000.
55. Markus Deserno, Axel Arnold, and Christian Holm. Attraction and ionic correlations between charged stiff polyelectrolytes. *Macromolecules*, 36(1):249–259, 2003.
56. A. Naji, A. Arnold, C. Holm, and R. R. Netz. Attraction of similarly–charged rods: Simulations and strong coupling theory. submitted to Europhys. Lett.
57. Markus Porto. Ewald summation of electrostatic interactions of systems with finite extent in two of three dimensions. *J. Phys. A*, 33:6211, 2000.
58. D. J. Langridge, J. F. Hart, and S. Crampin. Ewald summation technique for one-dimensional charge distributions. *Comp. Phys. Comm*, 134:78, 2001.
59. Axel Arnold and Christian Holm. Mmm1d. in preparation.
60. A. Bródka. Ewald type summation method for electrostatic interactions in computer simulations of a three-dimensional system periodic in one direction. *Chem. Phys. Lett.*, 2002.
61. M. Kawata and M. Mikami. Rapid calculation of two-dimensional ewald summation. *Chem. Phys. Lett.*, 340:157–164, 2001.
62. M. Deserno, C. Holm, and K. Kremer. *Molecular dynamics simulations of the cylindrical cell model*, volume 99 of *Surfactant science series*, chapter 2, pages 59–110. Marcel Decker, New York, 2001.
63. M. Deserno and C. Holm. Theory and simulations of rigid polyelectrolytes. *Mol. Phys.*, 100(18):2941–2956, September 2002.
64. R. E. Rosensweig. *Ferrohydrodynamics*. Cambridge Univ. Press, Cambridge, 1985.
65. Z. W. Wang and C. Holm. Estimate of the cutoff errors in the ewald summation for dipolar systems. *J. Chem. Phys.*, 115:6277–6798, 2001.
66. N. Boghossian, O. Kohlbacher, and H.-P.Lenhof. Ball: Biochemical algorithm library. Research report, Max-Planck-Institut für Informatik, Saarbrücken, 1999.
67. C. J. Lejdfors. Gismos home page, 1998. `http://www.teokem.lu.se/gismos/`.
68. W. F. van Gunsteren. Gromos96 homepage, 1996. `http://www.igc.ethz.ch/gromos/`.
69. S. J. Plimpton. Fast parallel algorithms for short-range molecular dynamics. *J. Comp. Phys.*, 117:1–19, 1995.
70. M. Nelson, W. Humphrey, A. Gursoy, A. Dalke, L. Kale, R. D. Skeel, and K. Schulten. Namd– a parallel, object-oriented molecular dynamics program. *Int. J. Supercomput. Ap.*, 10(4):251–268, 1996.
71. M. Pütz and A. Kolb. Optimization techniques for parallel molecular dynamics using domain decomposition. *Comp. Phys. Comm.*, 113:145–167, 1998.
72. M. Doi. Octa homepage, 2003. `http://octa.jp/OCTA/whatsOCTA.html`.
73. A. Arnold, B. A. Mann, H. J. Limbach, and C. Holm. Espresso - an extensible

simulation package for research on soft matter systems. *to be published*, 2004.
74. Tcl/Tk. Homepage, 2003. `http://tcl.activestate.com/`.
75. VMD. Homepage, 2003. `http://www.ks.uiuc.edu/Research/vmd/`.
76. R. Wunderling and M. Zöckler. DOC++: A documentation system for C/C++ and Java, 2003. `http://www.zib.de/Visual/software/doc++/`.
77. CVS. Concurrent versions system - homepage, 2003. `http://www.cvshome.org/`.
78. P. Ahlrichs and B. Dünweg. Simulation of a single polymer chain in solution by combining lattice boltzmann and molecular dynamics. *J. Chem. Phys.*, 111(17):8225–8239, 1999.
79. Pep Español and Patrick Warren. Statistical mechanics of dissipative particle dynamics. *Europhys. Lett.*, 30:191, 1995.
80. T. Soddemann, B. Dünweg, and K. Kremer. Dissipative particle dynamics: A useful thermostat for equilibrium and nonequilibrium molecular dynamics simulations. *Phys. Rev. E*, 68:046702, 2003.
81. Florian Müller-Plathe. Reversing the perturbation in non-equilibrium molecular dynamics: An easy way to calculate the shear viscosity of fluids. *Phys. Rev. E;*, 59:4894–4899, 1999.
82. A. C. Maggs and V. Rosseto. Local simulation algorithms for coulombic interactions. *Phys. Rev. Lett.*, 88:196402, 2002.
83. M. Hara, editor. *Polyelectrolytes: Science and Technology*. Marcel Dekker, New York, 1993.
84. Y. Kantor and M. Kardar. Excess charge in polyampholytes. *Europhys. Lett.*, 27:643, 1994.
85. Y. Kantor and M. Kardar. Instabilities of charged polyampholytes. *Phys. Rev. E*, 51:1299, 1995.
86. A. V. Dobrynin, M. Rubinstein, and S. P. Obukhov. Cascade of transitions of polyelectrolytes in poor solvents. *Macromolecules*, 29(8):2974, 1996.
87. U. Micka, C. Holm, and K. Kremer. Strongly charged, flexible polyelectrolytes in poor solvents - a molecular dynamics study. *Langmuir*, 15:4033, 1999.
88. H. J. Limbach and C. Holm. End-effects of strongly charged polyelectrolytes - a molecular dynamics study. *J. Chem. Phys.*, 114(21):9674–9682, 2001.
89. H. J. Limbach and C. Holm. Conformational properties of poor solvent polyelectrolytes. *Comp. Phys. Comp.*, 147:321–324, 2002.
90. H. J. Limbach and C. Holm. Single-chain properties of polyelectrolytes in poor solvent. *J. Phys. Chem. B*, 107(32):8041–8055, 2003.
91. H. Schiessel and P. Pincus. Counterion-condensation-induced collapse of highly charged polyelectrolytes. *Macromolecules*, 31:7953–7959, 1998.
92. A. R. Khokhlov. On the collapse of weakly charged polyelectrolytes. *J. Phys. A*, 13:979, 1980.
93. T. A. Vilgis, A. Johner, and J.-F. Joanny. Stretching necklaces. *Eur. Phys. J. E*, 2(3):289–300, 2000.
94. M. J. Stevens and Kurt Kremer. The nature of flexible linear polyelectrolytes in salt free solution: A molecular dynamics study. *J. Chem. Phys.*, 103(4):1669–1690, 1995.
95. M. O. Khan, S. M. Melnikov, and B. Jönsson. *Macromol.*, 32:8836, 1999.
96. C. Heitz, M. Rawiso, and J. François. X-ray scattering study of a poly (methacrylic

acid) sample as a function of its neutralization degree. *Polymer*, 40(7):1637–1650, 1999.
97. R. Schweins and K. Huber. preprint. *to be published*, 2004.
98. M. Nierlich, C. Williams, F. Boue, J. Cotton, M. Daoud, B. Garnoux, G. Jannink, C. Picot, M. Moan, C. Wolff, M. Rinaudo, and P. G. de Gennes. Small-angle neutron-scattering by semi-dilute solutions of polyelectrolyte. *J. Physique*, 40:701, 1979.
99. J. F. Joanny. Scaling description of charged systems. In C. Holm, P. Kékicheff, and R. Podgornik, editors, *Electrostatic Effects in Soft Matter and Biophysics*, volume 46 of *NATO Science Series II - Mathematics, Physics and Chemistry*, pages 149–170. Kluwer Academic Publishers, Dordrecht, Nl, December 2001.
100. D. Baigl, R. Ober, D. Qu, A. Fery, and C. E. Williams. Correlation length of hydrophobic polyelectrolyte solutions. *Europhys. Lett.*, 63(4):588–594, 2003.
101. W. Essafi, F. Lafuma, and C. E. Williams. Effect of solvent quality on the behaviour of highly charged polyelectrolytes. *J. Phys. II*, 5:1269–1275, 1995.
102. T. A. Waigh, R. Ober, C. E. Williams, and J.-C. Galin. Semidilute and concentrated solutions of a solvophobic polyelectrolyte in nonaqueous solvents. *Macromolecules*, 34(6):1973–1980, 2001.
103. A. V. Dobrynin and M. Rubinstein. Hydrophobic polyelectrolytes. *Macromolecules*, 32(3):915–922, 1999.
104. A. V. Dobrynin and M. Rubinstein. Counterion condensation and phase separation in solutions of hydrophobic polyelectrolytes. *Macromolecules*, 34(6):1964–1972, 2001.
105. H. J. Limbach, C. Holm, and K. Kremer. Structure of polyelectrolytes in poor solvent. *Europhys. Lett.*, 60(4):566–572, 2002.
106. C. Holm, H. J. Limbach, and K. Kremer. Poor solvent polyelectrolytes. *J. Phys.: Condens. Matter*, 15:S205–S211, 2003.
107. H. J. Limbach and C. Holm. *in preparation*, 2004.